UNIVERSIDADE DA BEIRA INTERIOR
Faculdade de Engenharia
Departamento de Informática

Content Storage and Retrieval Mechanisms for Vehicular Delay-Tolerant Networks

Bruno Miguel Correia Silva

Submitted to the University of Beira Interior in candidature for the Degree of Master of Science in Informatics Engineering

Supervised by Prof. Doutor Joel José Puga Coelho Rodrigues

Department of Informatics
University of Beira Interior
Covilhã, Portugal
http://www.di.ubi.pt

Acknowledgements

First of all, I would like to thank Professor Joel José Puga Coelho Rodrigues for giving me the chance to join his research group, Next Generation Networks and Applications Group (NetGNA), for all the constant words of encouragement and for supervising my Master's Thesis. This two years were undoubtedly very important in my evolution has informatics Engineer and person as well.

I am most grateful to the University of Beira Interior, the *Instituto de Telecomunicações*, Next Generation Networks and Applications Group (NetGNA), Covilhã Delegation, Portugal in the framework of the Project VDTN@Lab, and by the Euro-NF Network of Excellence from the Seventh Framework Programme of EU, for many kinds of all the support that was given to me.

Additionally, I'd like to thank to all members of the VDTN team, but especially to my colleagues João Isento and João Dias, for all the daily support and help. Special thanks to Vasco Soares for being always thoughtful and helpful. Last but not least, to my Mother, Father, Grandmothers and Girlfriend, the most important people in my life. Thanks for all the support and help in all the hard times. They are the "pillars" of my "foundation", all that I am and all that I have, I owe to them.

Abstract

Vehicular delay-tolerant networks (VDTNs) were proposed as a novel disruptive network concept based on the delay tolerant networking (DTN) paradigm. VDTN architecture uses vehicles to relay messages, enabling network connectivity in challenging scenarios. Due to intermittent connectivity, network nodes carry messages in their buffers, relaying them only when a proper contact opportunity occurs. Thus, the storage capacity and message retrieving of intermediate nodes directly affects the network performance. Therefore, efficient and robust caching and forwarding mechanisms are needed. This dissertation proposes a content storage and retrieval (CSR) solution for VDTN networks. This solution consists on storage and retrieval control labels, attached to every data bundle of aggregated network traffic. These labels define cacheable contents, and apply cache-control and forwarding restrictions on data bundles. The presented mechanisms gathered several contributions from cache based technologies such as Web cache schemes, ad-hoc and DTN networks. This solution is fully automated, providing a fast, safe, and reliable data transfer and storage management, while improves the applicability and performance of VDTN networks significantly. This work presents the performance evaluation and validation of CSR mechanisms through a VDTN testbed. Furthermore it presents several network performance evaluations and results using the well-known DTN routing protocols, Epidemic and Spray and Wait (including its binary variant). The comparison of the network behavior and

performance on both protocols, with and without CSR mechanisms, proves that CSR mechanisms improve significantly the overall network performance.

Keywords

Content Storage and Retrieval, Cache, Web caching, Mobile Ad-hoc Networks Caching, Vehicular Ad-hoc Networks caching, Delay-Tolerant Networks Caching, Vehicular Delay-Tolerant Networks.

Contents

List of Figures

List of Tables

Acronyms

BAD	:	Bundle Aggregation and De-aggregation
BSC	:	Bundle Signaling Control
CPU	:	Central Processing Unit
CSR	:	Content Storage and Retrieval
DTN	:	Delay Tolerant Networks
FIFO	:	First-in-First-out
Gbytes	:	Giga Bytes
ICP	:	Internet Cache Protocol
IDE	:	Interface Development Environment
IEEE	:	Institute of Electrical and Electronics Engineers
IP	:	Internet Protocol
ITS	:	Intelligent Transport System
Kbytes	:	Kilo Bytes
MANET	:	Mobile Ad-hoc Network
PDA	:	Personal Digital Assistant
SDK	:	Software Development Kit
TTL		Time-to-live
UML	:	Unified Modelling Language
VANET	:	Vehicular Ad-hoc Network
VDTN	:	Vehicular Delay-Tolerant Networks

1. Introduction

1.1. Focus

A weather station on top of a mountain collects data from various sensors types, such as, temperature, humidity, precipitation, and wind velocity, among others. Passing by vehicles equipped with an on-board computer collects data and distribute it through out the mountain on specific locations or to other vehicles. Vehicles disseminate data by transmitting it to other vehicles or kiosks that are in the vehicles route. This scenario perfectly illustrates an applicability of vehicular delay-tolerant networks (VDTN) [1]. VDTN appears as novel network architecture based on the concept of delay tolerant networks (DTN) [2], gathering also contributions from ad-hoc and vehicular networks, using vehicles for network connectivity.

Mobile ad-hoc networks (MANETs) [3] are based on autonomous ad-hoc networks where mobile hosts are connected by multi-hop wireless links. MANETs does not rely on common network infrastructures and services, the network nodes perform themselves all the networking functions [4]. Typical applications of MANET include rescue operations [5], military scenarios [6], and cases in which is impossible to establish a wired backbone [7]. This communication technology can also be used for extending the coverage of current wireless networks.

Vehicular ad-hoc networks (VANETs) [8] are a subset of MANETs and an emerging topic on intelligent transportation systems (ITS) [9]. The key difference between MANETS and VANETs is the mobility of the network nodes that can be anticipated due to the roads and traffic signalization (traffic lights) and also the rapid changes on the network topology due to the vehicles velocities [10]. VANETs does not rely on fixed infrastructures because vehicles provide all network services, themselves, relaying data between them. The VANET architecture also includes 'Infostations' that usually are wireless access points placed in specific points of the network and connected to the Internet. They provide high bandwidth connectivity to the Internet for vehicles that are passing by. Then, vehicles will deliver and propagate data through the network [11, 12]. This data may include information about collision alerting, road conditions warnings and also Internet connectivity for several applications [13, 14] such as, eMail, Web browsing, audio and video streaming, among others. These networks are usually found in several scenarios like notification of traffic condition (jams), accident warnings, advertisements [11], and cooperative vehicle collision avoidance [15]. It may also be used to gather information collected by vehicles, such as, road pavement defects [16]. VANETs are characterized by moving vehicles and their high mobility, assuming contemporaneous end-to-end path with continuous connection among devices. But due to the vehicles velocity and limited connections capacity, traditional routing protocols are inefficient and insufficient. Without a continuous connection, the communication data gets lost everytime that a connection is broken. To answer these intermittent connection problems and the delays intolerance, it's often used the DTNs [2].

A DTN enables communication in environments with high level of network disruption. DTN networks are presented as one solution to challenged environments characterized by intermittent connectivity, long and variable delays and with poor, sparse and none end-to-end connectivity [17]. DTN networks include storage-and-forward functions by overlaying the bundle protocol layer over the transport layer. Network nodes (mobile and fixed) relay these bundles between gateways, and intermediate nodes store and carry bundles until a proper contact is available [18]. The bundle protocol [19] aims to resolve several DTN issues, such as intermittent connectivity, long or variable delays and high error rates. DTNs have been applied to various areas like interplanetary networks [20], underwater networks [21], wildlife tracking sensor networks [22], and networks for developing communities [23]. DTN architecture defines different types of contacts classified as opportunistic, scheduled and predicted. Vehicular Networks are an example of opportunistic contacts.

VDTN [1] proposes to resolve the communication disconnection problem. VDTN is a particular application of a mobile DTN where vehicles are opportunistically exploited to offer a message relaying service. It intends to provide low-cost connectivity in scenarios where the telecommunications infrastructure is unreliable or not available due to disconnected areas, natural disaster, or emergency situations. VDTN is a challenging topic that faces several network issues and research challenges, such as, network architecture (naming and addressing), node design (power, storage capacity, range, speed, physical link), node type (mobile, stationary), node interactions, node cooperation, network topology (known or not), mobility pattern (deterministic, stochastic, predictable, etc), scheduling, traffic (static, dynamic), routing protocols, bundle format, caching mechanisms, security, and supported applications.

This novel network architecture also adopts a DTN store-carry-and-forward paradigm. However, it distinguishes itself from the DTN architecture by positioning the bundle layer between network and link layers. Network nodes (fixed or mobile) store data on their buffers waiting for new contact opportunities to forward data to any node that establish a proper communication opportunity. Figure 1, illustrates the interactions between nodes. Terminal nodes represent the access points to the VDTN network. Mobile nodes (e.g. vehicles) collect and disseminate traffic data. Relay nodes are fixed devices with store and forward capabilities and are located at road intersections.

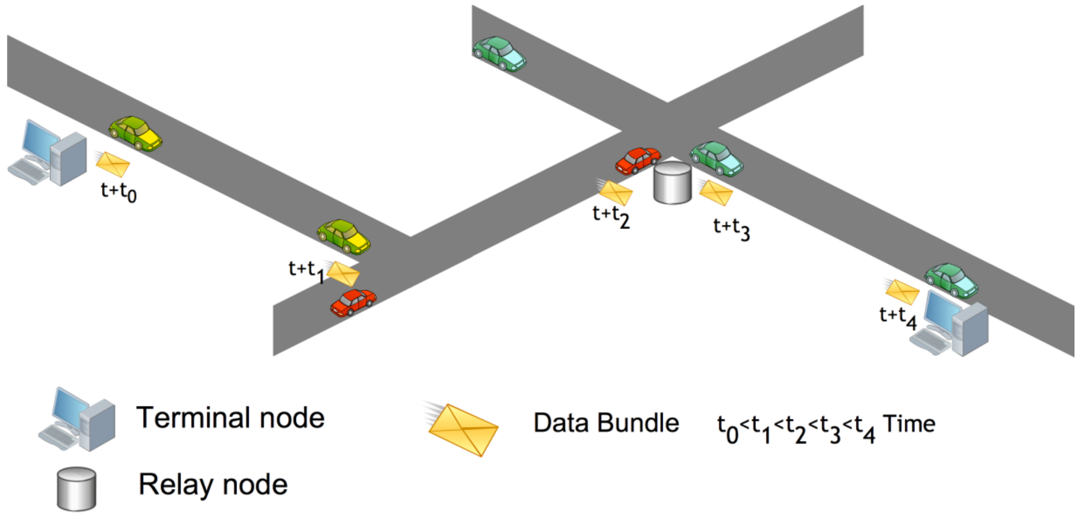

Figure 1. Store-carry-and-forward paradigm of DTN-based networks.

To address the problem of intermittent connectivity, network nodes store messages on their buffers, carrying them through the network waiting for new transfer opportunities. Thus the storage capacity of the nodes directly affects the performance of the network. Therefore it's important the incorporation of suitable network protocols using self-contained messages to improve communication that supports store-carry-and-forward operation procedures. Clearly, such procedures define the act of content storage and retrieval (CSR).

This dissertation presents a CSR solution for VDTNs, fully automated, that manages the nodes buffer and decides which data is to be stored, sent or eliminated. This solution provides a fast, safe, and reliable storage management and data transfer, improving the applicability and the performance of VDTN networks significantly. This intelligent caching and forwarding solution mainly consists in storage and retrieval control labels attached to every data bundle. The main purpose of the *VDTN CSR labels* is to define cacheable contents and apply them cacheable and forwarding restrictions. CSR mechanisms gathered contributions from various cache-based technologies, namely, Web-caching, ad-hoc networks caching mechanisms, and DTNs storage and forward schemes. These approaches have similar network architectures or similar used technologies as the VDTN network architecture offering various contributions to the storage and forward proposal for VDTNs. Furthermore, this work presents the performance assessment and validation of the CSR proposal. This evaluation proves its feasibility and also presents the impact and analysis of the proposed CSR mechanisms on the performance of the VDTN network, through a VDTN testbed. The testbed scenario considers the use of the well known DTN Epidemic and Spray and Wait (and its binary variant) routing protocols, applied to VDTNs. For comparison purposes, a network scenario with and without CSR mechanisms were considered for node's bundle delivery probability, bundle average delay, and average contact duration.

1.2. Objectives

The main objective of this dissertation is to present a robust solution to provide a fast, safe and reliable content storage and retrievable management, improving significantly the applicability and performance of the VDTN network. To accomplish this main objective, the first major task is the revision on VDTN literature and review of the state of the art on caching and forward related projects. This review focuses on Web-caching schemes, ad-hoc networks caching mechanisms, and mainly on DTNs storage and forward solutions. The next step is the design and implementation of the CSR mechanisms for VDTNs. Beginning with its conceptual design and the construction of data control algorithms responsible for caching and forward operations. These operations define the intelligent cashing solution needed to improve the performance of our VDTN network. With the CSR proposal implemented and validated, a VDTN testbed is used for network performances and evaluations. These performance assessments prove the feasibility of the CSR mechanisms, presenting the impact and analysis on the performance of the VDTN network.

1.3. Main Contributions

This section is devoted to the scientific contributions of this dissertation to the state-of-the-art on VDTNs and caching and forward mechanisms. The main contribution is a content storage and retrieval proposal for VDTNs and its performance assessments through a VDTN testbed, which is presented in Chapters 3 and 4. This proposal was accepted for presentation at IEEE GLOBECOM 2010, Miami, Florida, USA, December 6-10, 2010.

1.4. Dissertation Structure

This dissertation is organized in 5 chapters. This chapter, the first, presents the context of the dissertation, focusing on the topic under study, the objectives, the main contributions and the dissertation structure. Chapter 2 introduces the VDTN concept and architecture describing its key features and main contributions. It elaborates on the related work about the topic, focusing on Web-caching, ad-hoc networks caching schemes and DTNs storage and forward mechanisms. Afterwards, it presents the CSR proposal for VDTN. Chapter 3, describes the proposed VDTN CSR mechanisms including its conceptual design, used technologies and presents the VDTN testbed used for the evaluation and validation of the CSR proposal. Chapter 4, focuses on the performance evaluation and validation study of the CSR mechanisms through the VDTN testbed. Finally, Chapter 5 concludes the dissertation and proposes topics for future research works.

2. Content Storage and Retrieval Mechanisms

This chapter elaborates on content storage and retrieval (CSR) approaches available in the related literature and proposes an approach of CSR mechanisms proposal for VDTN networks. It begins by introducing the VDTN concept and architecture, including its key features and main contributions. Afterwards, the state of the art on caching and forwarding related projects is presented, namely on Web-caching, ad-hoc networks caching schemes and DTNs storage and forward mechanisms. These approaches have contributed to the construction of the CSR proposal. Finally, it proposes and describes the CSR proposal for VDTN, introducing its main features and behaviors.

2.1 Vehicular Delay-Tolerant Networks

VDTN architecture handles non-real time applications where vehicles act as the communication infrastructure for the network. A typical VDTN architecture is shown in Figure 2. It includes three types of node, *terminal nodes*, *relay nodes*, and *mobile nodes*. *Terminal nodes* act as access points to the VDTN network providing connection to end-users. It is assumed that at least one *terminal node* has direct access to the Internet, providing several services including exchanging emails, documents, voice mails, movie, music, images etc…. *Mobile nodes* (e.g., vehicles) are responsible for physically carrying data between *terminal nodes* located at isolated points of the network. *Relay nodes* are devices with store-and-forward capability allowing bypassing *mobile nodes* to drop and pickup data. These nodes increase the number of contact opportunities in sparse scenarios, contributing to improve the delivery ratio and decrease the delivery delay. In a low node density scenario, relay nodes increase contact opportunities [24, 25]. Therefore, they contribute to improve the data bundle delivery probability and decrease their average delivery delay. The VDTN architecture has been successfully tested and validated through a creation of a VDTN testbed [26].

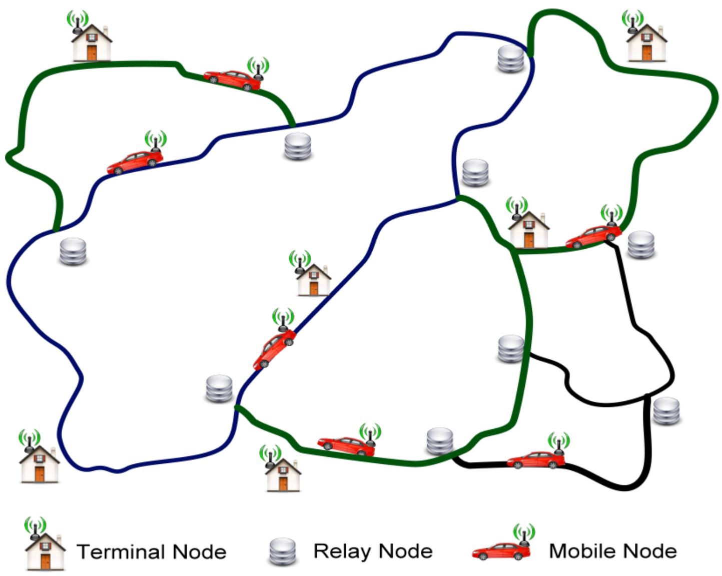

Figure 2. An Illustration of typical VDTN architecture.

Contrary to DTN architecture proposal, witch introduces a bundle layer between the transport and application layer, for storage and forward services the VDTN architecture proposes a bundle layer placement over the data link layer introducing an IP over VDTN approach (Figure 3). The protocol data unit at the VDTN bundle layer is called a bundle. A bundle is an aggregation of incoming IP datagrams into data bundle messages. This proposal is a key contribution of the VDTN architecture resulting in less and more fast processing and therefore an efficient energy saving [1].

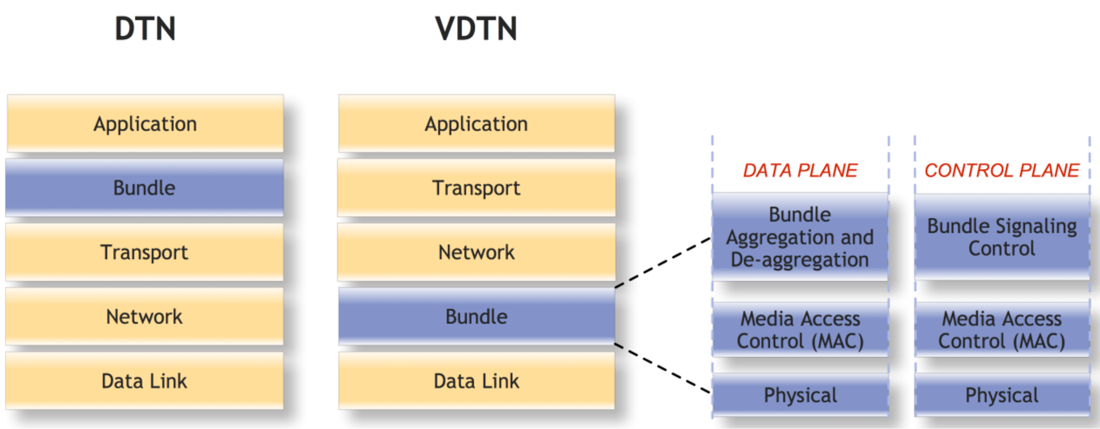

Figure 3. DTN and VDTN layered architectures.

VDTN makes use of store-carry-and-forward paradigm proposed for DTNs. This paradigm solves the problems caused by intermittency, disconnection and long delays, and can be described as follows. Using some form of persistent storage, a network node stores a bundle, while waiting for a future proper connection opportunity. When a communication opportunity occurs, the bundle is forwarded. This process is repeated and bundles are relayed hop-by-hop until eventually they reach the destination.

VDTN architecture identifies two logical planes, i.e., control plane, and data plane, which are illustrated in Figure 3. Hence, the VDTN bundle layer is logically divided into two layers, the bundle signalling control (BSC) layer, and the bundle aggregation and de-aggregation (BAD) layer. The BSC layer executes the control plane functions such as signalling messages exchange, resources reservation (at the data plane) and routing. Signalling messages are used to setup a data plane connection and carry information about a node type and its speed, physical link data rate and range, energy constraints, storage capacity constraints, delivery options, security requirements, among others. The BAD layer executes the data plane functions. These functions include storage management, queuing and scheduling, traffic classification, among others. Control plane functions include an out-of-band signaling which uses a low-power, low bandwidth, long-range link and its always active to allow node discovery. At the same time, the data plane is only active during the estimated contact duration, if there are data bundles that should be exchanged between nodes. Otherwise, the data plane link connection is not activated [1, 27]. This approach is very important for optimization of the available data plane resources (e.g. storage and bandwidth), and also allows saving power, which is very important for energy-constrained nodes such as relay nodes that usually run on solar panels or batteries [1, 28]. Figure 4, illustrates this concept. At the time $t+t_0$, a mobile node and a relay node detect each other and start exchanging control information through the control plane link connection. Through the routing information, both nodes determine which bundles should be forwarded. Then, the data plane connection is configured and activated on both nodes at the time $t+t_1$. Data bundles are exchanged until the time $t+t_2$. At the instant that nodes leave the data plane link range of each other, the data plane connection is deactivated. The separation of control and data planes is a key contribution of the VDTN layered architecture. This concept is similar to the principle presented in [29] for Optical Burst Switching networks.

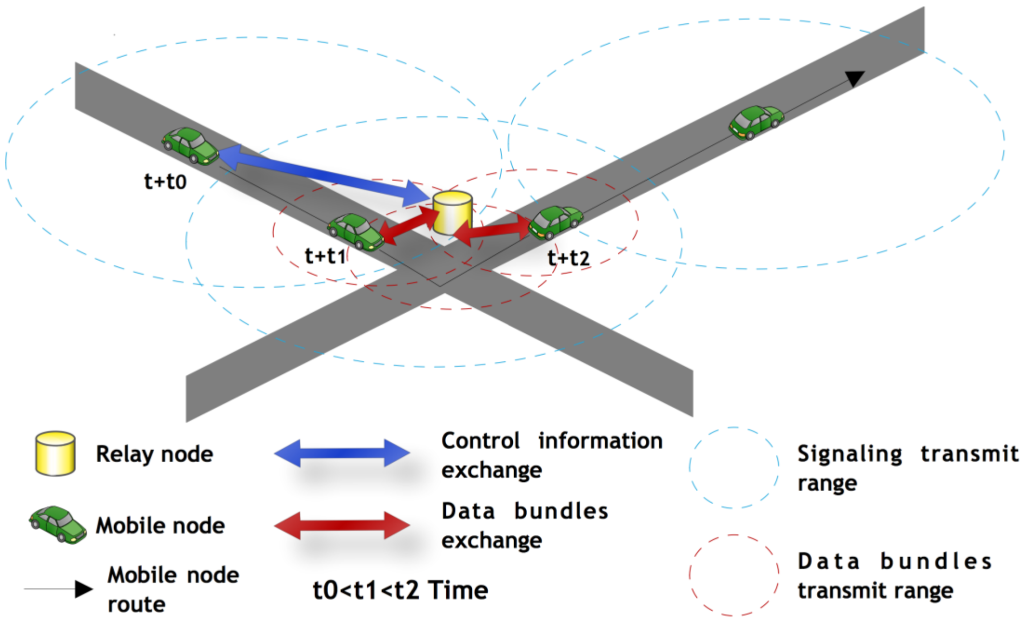

Figure 4. Control information and data bundles exchange.

2.2 Caching and Forward Related Approaches for VDTNs

Several cache-base approaches such as, Web-caching, ad-hoc networks caching schemes and DTNs storage and forward mechanisms, addresses analogous issues to VDTN networking (i.e., terminal's latency, connectivity, energy constraints and overall network performance). The study of these caching and forward related approaches has offered several contributions to the design of the CSR solution applied to the VDTN architecture.

2.2.1 Web Caching Mechanisms and Schemes

Over the last years several types of networks but especially the World Wide Web (WWW) have exponentially grown the number of its users. The convergence of wireless communications and the Internet results in mobile Internet, where users easily access the Web almost, anytime and anywhere. These conditions result in bigger user latency and network traffic congestion motivating the implementation of intelligent caching schemes. Caching mechanisms are applied on several network architectures and communication technologies in order to reduce latency, reduce bandwidth usage and server load. Cache in its definition is basically a fast and temporary storage buffer. Duplicated cached data can be accessed in the future rather than the original data, saving computational resources. An analogous technology is a proxy server [30]. A proxy server act as intermediate for a client request and seeks the response in other servers or in a local cache. A caching proxy server improves the service by providing a fastest response and retrieving content stored in previous requests [31]. Figure 5, illustrates a generic web caching system including proxy cooperation.

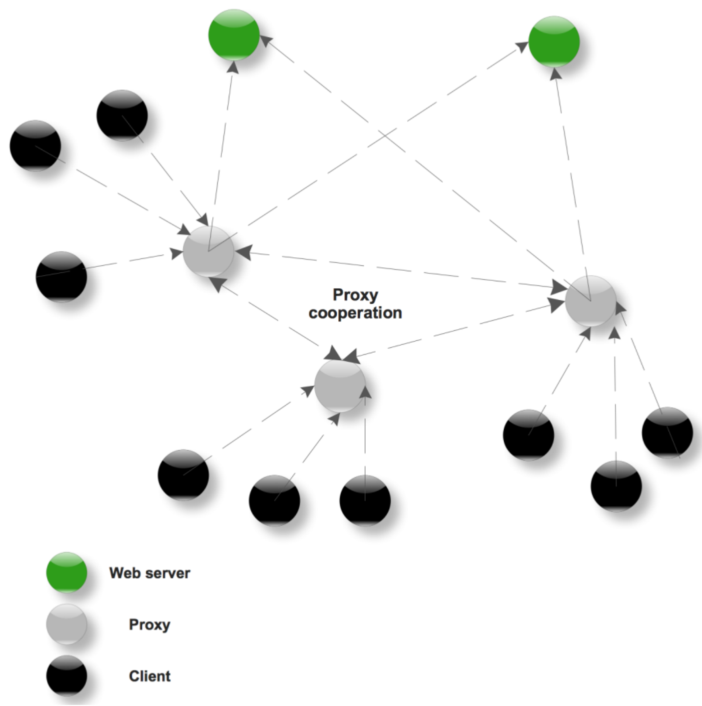

Figure 5. A Web caching scheme with proxy cooperation.

Web caching protocols have proven to be the ideal solution to these specific network problems [32]. Thus was born the concept of network cooperative cache [33]. Web cooperative cache schemes are used by many public organizations (e.g. *IRCache* [34], *NLANR* [35]). Cooperative cache is based on a hierarchical approach similar to the *Internet Cache Protocol* (ICP) [36]. The first hierarchical approach was proposed in the *Harvest project* [37]. Other related approaches are *Access Driven cache* [38], *Adaptive web caching* [39], *Cachemesh* [40], *Geographical push caching* [41], etc… Cooperative cache schemes basically introduce a cache on every network node. Afterwards all caches emit a request for missing objects to a cache at its hierarchy's upper level. This process is repeated until the request is found or it reaches the roots cache. Finally, as it returns to the requester, the object is copied in all caches.

2.2.2 Caching Schemes in Ad-Hoc Networks

Ad-hoc networks [42] consists on wireless networks nodes cooperating to forward traffic between themselves. Ad-hoc networks topology raises several fundamental research issues that are still being debated. One of these issues is the most appropriate caching strategy, since network nodes have limited store capability and several energy constraints. In [33] is proposed a caching solution to improve mobile performance focusing on terminal's latency, connectivity and energy constraints. Upon a request, each node uses local statistics of every interaction that identify mobile terminals that have most likely stored the requested data. It also uses a list of known mobile terminals and a routing table in order to find all nodes that are in inferior range to the next base station. Afterwards mobile terminals are contacted from the closest one to the farthest, until a hit message is received or there are no more mobile terminals to contact. To minimize network load and energy consumption, this approach uses timeouts instead of miss messages to notify mobile terminals that a request object was not cached.

A cooperative cache protocol for ad hoc networks was proposed in [43]. It presents an efficient approach to save energy consumption and minimize network load. The proposal presents three distinct cache protocols: *CacheData*, *CachePath* and *HybridCache*. In the *CachePath* protocol, when two nodes communicate they exchange requesters' node identification related to data that they carry. It also caches the distance between themselves and the source and destination (path) of the requested data. Therefore, when another node issues the same request, the request will be redirect to the node with the *id* cached previously. In the *CacheData* protocol, instead of paths, router nodes frequently caches accessed passing-by data for future requests. The *HybridCache* protocol use both *CacheData* and *CachePath* protocols.

Marco Fiore et al proposes in [44] a cooperative caching algorithm called *Hamlet*. This caching strategy works independently on each network node deciding which contents are cacheable and for how long. These caching decisions are taken according to a probabilistic estimate of what nearby nodes may have stored on their caches and with the intention of differentiate its own stored content from the others. *Hamlet* can be applied to any content distribution system where network nodes can overhear query/response messages, estimate its distance in hops from the query source node and the responding node, identify the version of the transmitting information and associate a specific drop time to each item on its cache. Cooperative Web cache strategies are also applied to mobile terminals that interact via ad-hoc networks.

MANETs are autonomous Ad-hoc networks consisting on mobile hosts connected by multi-hop wireless links [3]. MANETs network architecture raises several research issues such as, intermittent network connections, power supply and limited computing resources. These issues directly affect the data availability and access efficiency of network nodes [45]. Cooperative cache solutions help saving energy, time and bandwidth by localizing communications, improving data availability and access efficiency. A cooperative caching service for MANETs is presented in [45] and illustrated in Figure 6. This proposal addresses two important issues in cooperative caching - cache resolution and cache management. Cache resolution concerns the decision mechanisms of the mobile device to find the requested data from the user. For this purpose, historical profiles and forwarding nodes are used to induce less communication cost. Cache management determines which data will be placed on/purged from the local cache. This management decreases significantly the number of cached copies between nodes, allowing a bigger distinctive data storage and improving the network performance [45].

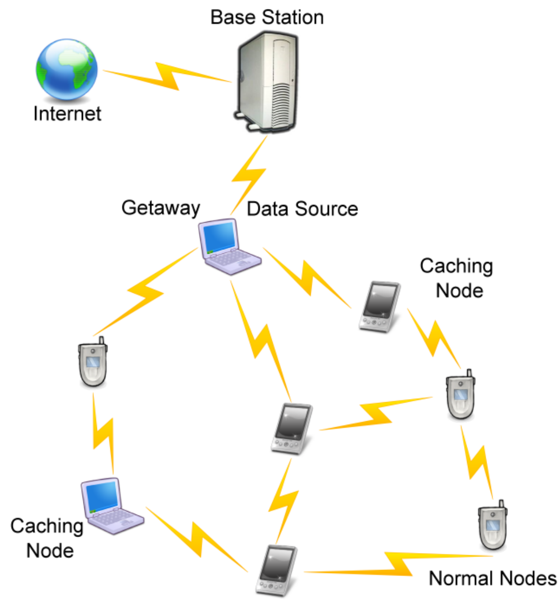

Figure 6. A cooperative caching scheme for MANETs.

A cooperative caching scheme for MANETs is also proposed in [46]. This proposal includes a distributed application software that implements ad-hoc cooperative Web caching among mobile terminals. This approach aims to improve Web latency among mobile terminals, optimizing their energy consumption and also to increase data accessibility. It introduces a local caching strategy that is adaptive according to the capacity of the terminal. Every cached document is weighted according to its probability of being retrieved in the future and the energy cost associated with getting remotely the document. Documents with the lowest weights are removed from cache. The value of a document weigh is computed according to its *Popularity*, *AccessCost* and *Coherency* values. *Popularity* value indicates the probability of future access according to the number of times the document has already been requested since it has been cached. *AccessCost* value estimates the energy cost of getting the document remotely in case of being removed from cache. *Coherency* value balances the lifetime of a document and the energy remaining on the terminal. This value favors energy saving over the accuracy of the document.

Vehicular ad-hoc networks (VANETs) are a subset of MANETs where vehicles deliver and propagate data throughout the network. Due to network architecture similarities, caching and retrieval mechanisms in VANETS are very similar or identical to the ones used in MANETs [8]. Ott et al presents in [47] the *Drive-thru Internet* project architecture. The *Drive-thru Internet* project consists on the use of IEEE 802.11 [48] access points that provide several Internet services for moving vehicles in challenged environments with intermittent connectivity. To resolve the disconnection problem this approach introduces the concept of *Performance Enhancing Proxies* (PEPs). PEPs [49] are usually network agents designed to improve end-to-end communication, breaking one connection into multiple connections. PEPs are commonly used to improve Transport Control Protocol (TCP) performance over a satellite link. *Drive-thru enhancing proxies* are placed in a fixed network managed by a *Drive-thru service provider* and are used as fixed points to relay mobile node requests to and from the Internet. Afterwards, they cache the requested data (e.g Mail services, web browsers) waiting for a new communication opportunity to forward it [13].

2.2.3 Caching Mechanisms in DTN networks

DTN network architecture is characterized by intermittently connected links and long variable latency. The bundle protocol uses persistent storage to resolve most of these issues [19]. DTN nodes store data permanently in a local storage unit. This data is transmitted when a contact opportunity occurs and in case of transmission failure it may be retransmitted several times. Since data must be moved from buffer to the storage unit and again back to buffer, persistence storage can result in bigger processing delays and therefore increasing the node's connection times. Since end-to-end path between client and server may not exist, adequated caching and retrieval mechanisms become even more important. One example where DTNs are perfectly adequated is in highly disconnected networks such as largely-disconnect villages. Cooperative caching and data prefetching techniques in such scenario are presented in [50]. This proposal addresses the usability of limited computer networks in remote villages, concerning webpages requests and specialized education applications. This approach uses *collaborative caching* allowing computers within a village to access webpages stored on other connected machines. It also introduces a *predictive prefetching* technique that delivers to a client or even to a village, webpages that have not yet been requested on the assumption that they will be requested in a near future. The advantage of this approach is that storage functions do not rely only on one single proxy server eliminating single points of failure and optimizing significantly the power consumption. Ott et al [51, 52] proposes a distributed caching scheme for DTN including *application hints* to messages to perform intelligent caching, to act as distributed storage or to forward content. The authors present application scenarios for DTN applications like HTTP [53] or email [23], considering *content retrieval* and *caching* scenarios and *content storage and retrieval* scenarios. Figure 7, illustrates the *content retrieval and caching* scenario, a user acts as a requester. The request is assumed to be a web resource. The user sends a request (*Req*) for resource (*U*), which travels through mobile nodes until it reaches a server. The server will then send a response (*Resp*) that travels all the way back to the user.

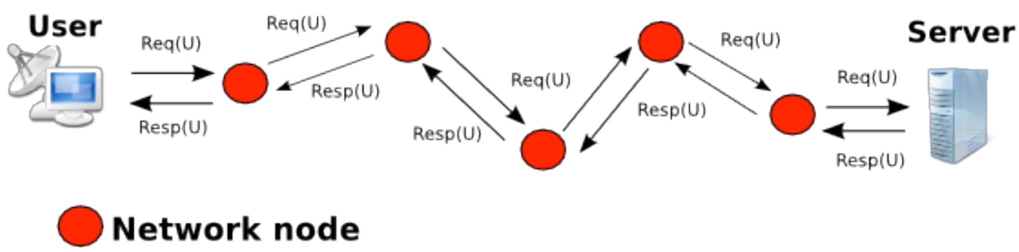

Figure 7. Illustrations of request and response operations between user and server.

On its way to retrieve the response (*Resp(U)*), the resource will be cached on every node until it reaches the requester. The request or the response may be replicated in various mobile nodes depending on the used routing algorithm. The replica stays stored on the node until the time to live (TTL) expires or the response gets to the requester or a more recent response replaces it. If another user issues a request for the same resource (*Req2 (U)*), this request may cross nodes that have already a replicated response (*Resp*). In this case the node will create a new response (*Resp2 (U)*) from its queued (*Resp (U)*) and replies it to the user, acting as an implicit cache. Therefore, nodes must be able to identify the resource to match both requests *(req2 (U))* and (*Resp1 (U))* and determine if the response is still the most recent one. This principle is illustrated in Figure 8.

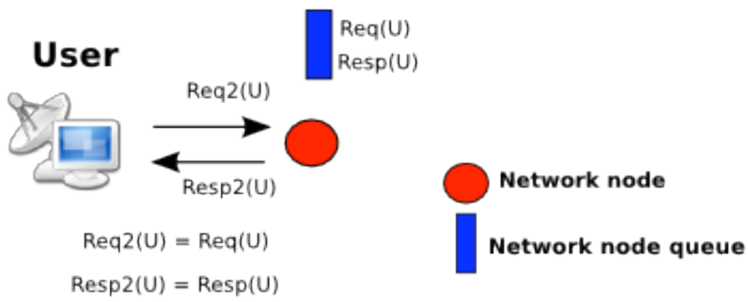

Figure 8. Illustration of request and response between user and a mobile node.

In the *content storage and retrieval* scenario, a user requests storage (*streq*) of a resource (*U*) to a server (Figure 9). The storage request (*streq (U)*) travels trough the nodes until it reaches the server. On its path, the resource (*U*) will be replicated along the nodes. While stored/queued, the resource *(U)* is available to any requester that is interested in it. The nodes must identify the local copy of the resource and create the response from previous storage requests (*streq (U)*). This mechanism may act as data backup for stored resources, in case of any hardware loss or failure.

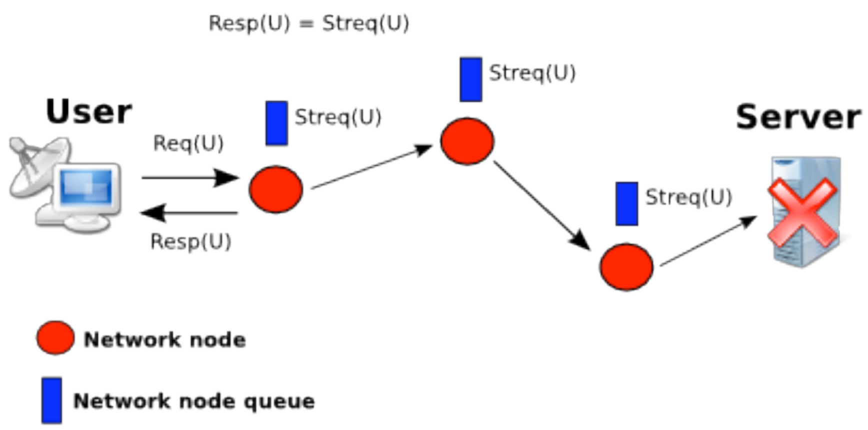

Figure 9. Illustration of storage operations on mobile nodes can perform backup services in a server failure scenario.

These scenarios clearly present a distributed caching strategy in DTN nodes. In order to improve this distributed caching scheme, the authors proposed in [54] a logic module for storage and retrieval operations. This module allows cache lookups for stored resources in the queue and cache retrieval for passing requests. Such procedures require additional information about their bundle payloads. For this purpose *application-hints extension blocks* [52] for the bundle protocols were also presented. The *application-hint* illustrated in Figure 10, contains several cache control fields. Among others, the *Application protocol* field shows the application protocol to perform proper resource matching. *Resource identifier* identifies the carried resource. *Operation type* indicates if the resource is a request, a response, an unconfirmed event, or an unknown operation type. These distinctions allow different storage policies. In case of bundle fragmentation, the application hint must be copied to all fragments.

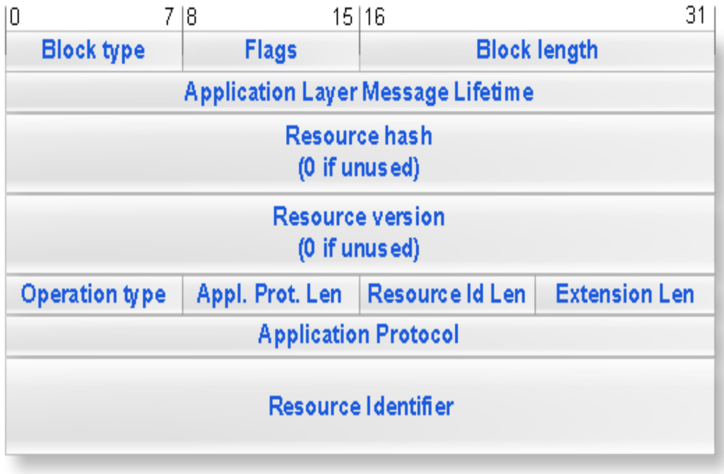

Figure 10. Illustration of an Application-Hint extension block for the bundle protocol.

2.3 Content Storage and Retrieval Proposal for VDTN

The VDTN CSR proposal assumes the inclusion of CSR control labels attached to every bundle of aggregated network traffic (data bundles). When network nodes interact, they first exchange control information, e.g. a setup message that furnishes information such as, among others, about bundles that they are carrying and their storage and energy constraints, node coordinates, velocity and destination. The *bundle CSR label* is sent also in the control channel and it is attached to the bundle setup message, as shown in Figure 11. The main purpose of these labels is to define cacheable and retrieval contents, and apply them caching and forward restrictions in order to improve network performance significantly, through elimination of all redundant information. Moreover, it also improves data transfers between nodes (data channel).

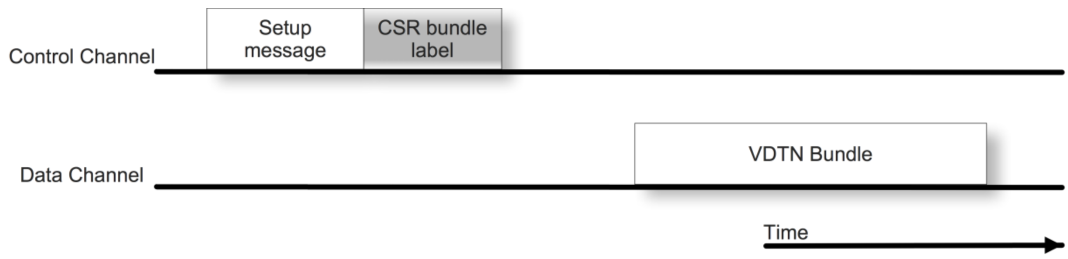

Figure 11. VDTN CSR bundle label is sent in the control channel attached to the Setup Message.

A VDTN data bundle is created by an aggregation of network IP traffic. Network traffic has different sources, sizes, destinations, and priorities. These properties directly affect the nodes buffer management, and therefore, the network performance. For each VDTN data bundle of aggregated network traffic, two different CSR control labels were created (Figure 12). The *CSR Bundle label* is applied to the VDTN bundle and contains for all aggregated network contents, their respective *CSR bundle fragment label*. The *CSR bundle fragment label* is applied to a single bundle fragment that contains network traffic.

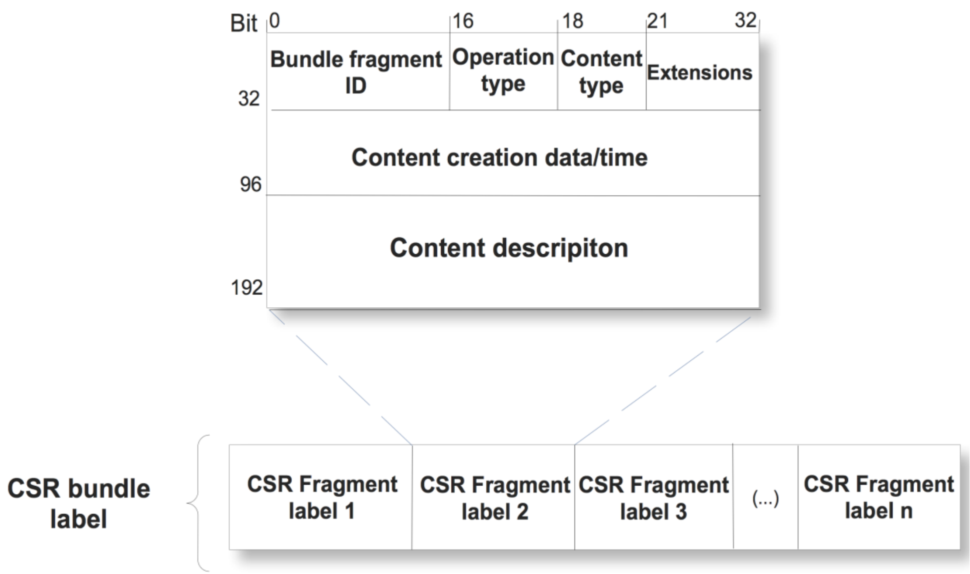

Figure 12. Illustrations of CSR bundle label and CSR fragment labels scheme.

For each bundle and fragment a CSR control label is applied. A CSR bundle label contains at least one CSR bundle fragment label. Each fragment *label* contains specific control fields that define if the content is to be cached or retrieved, as can be seen and described in Table 1. These control fields are the basis of the CSR operations.

Table 1. CSR CONTROL LABEL FIELDS

CSR control field	Description	Size
Bundle fragment label ID	Identifies the bundle label	16 bits
Content creation data/time	Data and the time of creation or last modification of the respective content	64 bits
Operation type	Specifies the operation for the respective content (i.e., request, response, storage)	2 bits
Content type	Type of the respective content (i.e., text, image, video)	3 bits
Content description	Describes the bundle content (i.e., www.content.com, image.png, sensordata32.txt, file.mp3)	96 bits
Extensions	Future extension fields	11 bits

3. Design and Application of CSR Mechanisms on VDTNs

This chapter presents the design and implementation of the CSR proposal for the VDTN architecture. It starts presenting the conceptual design of the CSR proposal, presenting UML diagrams of the main actions and procedures of a VDTN node with CSR mechanisms. The main storage and retrieval procedures are also presented including pseudo code of its key operations. Afterwards it presents the used technologies for the implementation of the CSR proposal. Finally, presents and describes the VDTN testbed used for implementation of CSR mechanisms.

3.1 Conceptual Design of CSR mechanisms

Stationary and mobile nodes share the same control and data plane, so CSR procedures will also be similar on all network nodes. Since vehicles are the network infrastructure of VDTN, the following UML [55] diagrams present the actions of a VDTN mobile node and its CSR mechanisms. The use case diagram of a VDTN mobile node and its control plane functions including the exchange of *CSR bundle labels* is presented in Figure 13. While traveling along its path, a mobile node searches for other network nodes using its control plane functions. This connection is always active and its operations are always under execution. When a mobile node finds another node, and a proper contact opportunity is established, both exchange control information. In this control information, along with the signaling message (setup message), CSR bundle labels are exchanged. The setup message information is used to decide whether this contact opportunity should be considered or ignored, and to configure data plane connection. If a data plane transmission is possible, the node executes the CSR mechanisms deciding which data is to be stored or retrieved.

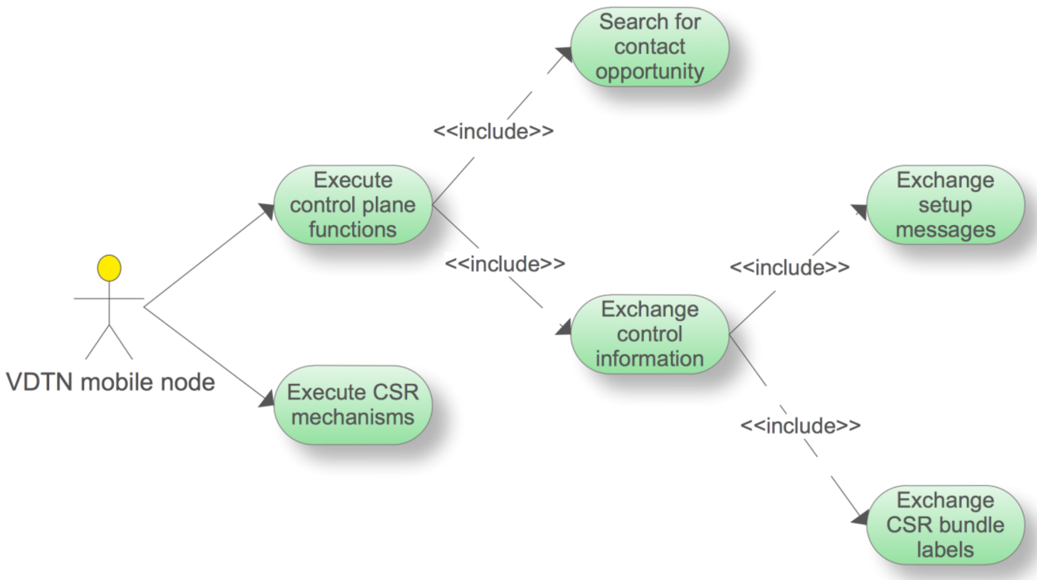

Figure 13. Use case diagram of a VDTN mobile node and its control plane functions including the exchange of CSR bundle labels.

The activity diagram of a VDTN mobile node, and its CSR mechanisms is presented in Figure 14. Through the signaling message, a contact opportunity is considered and the data plane is to be configured. Therefore, the mobile node starts its CSR mechanisms. After exchanging CSR bundle labels the node starts the CSR matching procedures. Through these matching operations, the node selects cacheable and forwarding data bundles to be sent and received from the other node. As soon as the matching process ends, data bundles that are sent as cacheable or retrievable are selected and exchanged when the data plane is connected and ready for data exchange.

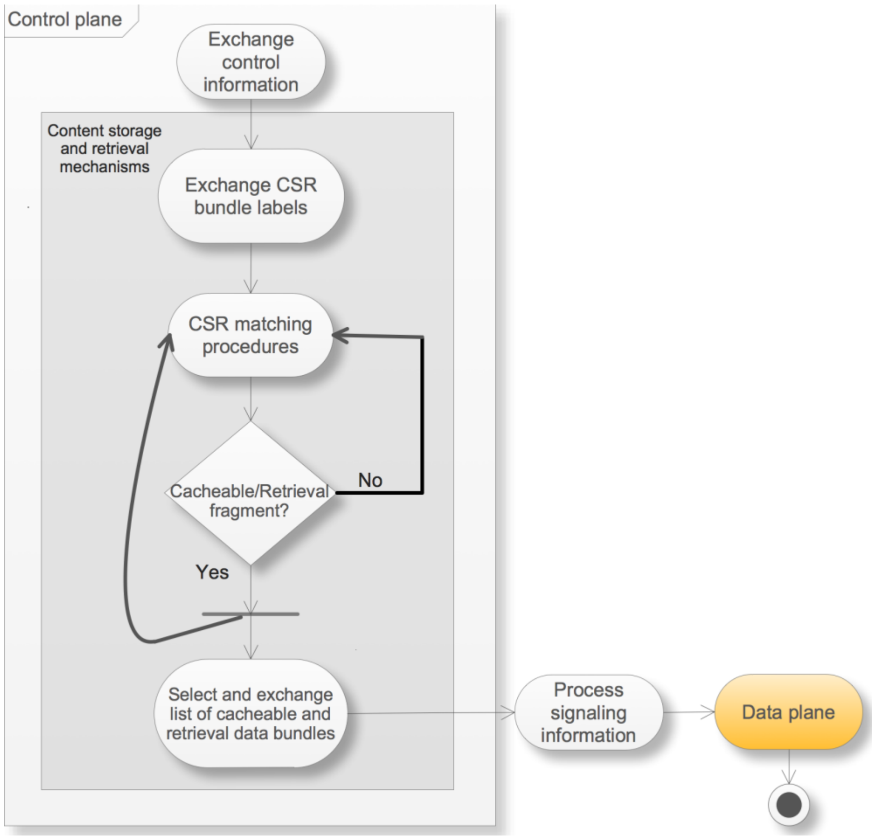

Figure 14. Activity diagram of a VDTN mobile node, and its CSR mechanisms.

3.1.1 Storage and Retrieval Procedures

CSR mechanisms allow that a given node may select bundles to be considered for storing and/or retrieving. After nodes exchange their *CSR bundle labels*, each one has access to the list of bundle fragments that the other node is carrying. A node only process bundles containing fragments that have '*Storage*', '*Response*', and '*Request*' as content '*Operation type*'. *Storage* type contents, are automatically selected for being stored, comparing only if a bundle with the same content is not already stored in the buffer node. The basic pseudo code used for implementing this matching principle is presented as follows.

```
/*Terminology
CSR_BL = CSR Bundle Label
CSR_FL = CSR Fragment Label
*/
1.  BEGIN
2.  Foreach CSR_FL in CSR_BL
3.  Foreach Operation_type in CSR_FL
4.  Read Operation_type
5.  if  Operation_type is Storage and CSR_FL.Content is not stored
       Add to store/receive list (CSRL_BL)
    else
         if Visitor_node_type is mobile node and
            Visitor_node_route includes CSR_BL.Destination
                   Add to store/receive list (CSR_BL)
                   else
                          continue.
6.  END
```

If the bundle has *responses* or *requests*, first it compares if a possible corresponding *request* or *response* is stored. If *yes*, the bundle is selected to be stored or retrievable (sent or received). If *no*, but at least one of the communicating nodes is mobile, the destination node is matched with the mobile node route information, comparing the nodes that it already visited and may visit. In this case, the bundle is selected to be stored or retrievable so that mobile node carries it to the destination. When CSR negotiation ends, all selected bundles are exchanged between nodes in a list of stored and retrieved data, as well as their respective CSR bundle labels. The basic operations for these matching principles in pseudo code are presented as follows.

```
/*Terminology
CSR_BL = CSR Bundle Label
CSR_FL = CSR Fragment Label
*/
1. BEGIN
2. Foreach CSR_FL in CSR_BL
3. Foreach Operation_type in CSR_FL
4. Read Operation_type
5. If Operation_type is Response
       if corresponding Request is stored
              Add to store/receive list (CSR_BL)
   else
    if Visitor_node_type is mobile node and
        Visitor_node_route includes CSR_BL.Destination
              Add to store/receive list (CSR_BL)
              else
                     continue.
6. if  Operation_type is Request
       if corresponding Response is stored
                     Add to store/receive list (CSR_BL)
   else
    if Visitor_node_type is mobile node and
        Visitor_node_route includes CSR_BL.Destination
              Add to store/receive list (CSR_BL)
              else
                     continue.
7. END
```

3.2 Used Technologies

For construction and implementation of the above-described CSR operations, it was used C# language. C# programming language [56], and deployed in the network nodes. They were developed using the .Net Compact Framework version 3.5 [57] for running in the PDA's Windows Mobile 6.1 [58], and using the .NET Framework for running in the desktop's and laptop's Windows Vista operating system. The .Net Framework contains a collection of libraries, which allow developers to run applications and services in Windows Platform. Microsoft Visual Studio 2008 Professional Edition [59] is a powerful IDE that contains a lot group of applications developed by Microsoft Corporation, to software development. Supports C#, J#, Visual Basic, and others.

3.3 VDTN testbed

A VDTN testbed [26] was used for performance evaluation and analysis of VDTN network with and without CSR mechanisms. This testbed implements the VDTN architecture demonstrating the applicability of VDTNs in several scenarios (e.g. interconnection of isolated regions) [26]. Figure 15, illustrates an activity diagram of a mobile node, which represents a workflow of stepwise activities and actions describing control plane and data plane interaction, coordinated by the decision module. This activity diagram is equal for all network nodes, and represents the concept of control plane and data plane separation with out-of-band signaling. Each network node autonomously manages its control plane and data plane link connections. Nodes are always searching for new contact opportunities, using their control plane link connection (low-power, low bandwidth, long-range), which is always active. A decision module is responsible for processing the control information exchanged at a new contact opportunity to decide whether to accept the contact, and for determining the amount of time that lasts the contact [27]. Then, the data plane link connection (high-power, high bandwidth, short-range) is activated, and remains in this state only during the estimated period of time that lasts the contact. Nodes use its data plane link to exchange data bundles. The bundle signaling control layer executes the control plane functions, such as, signaling messages exchange, resources reservation (at the data plane) and routing.

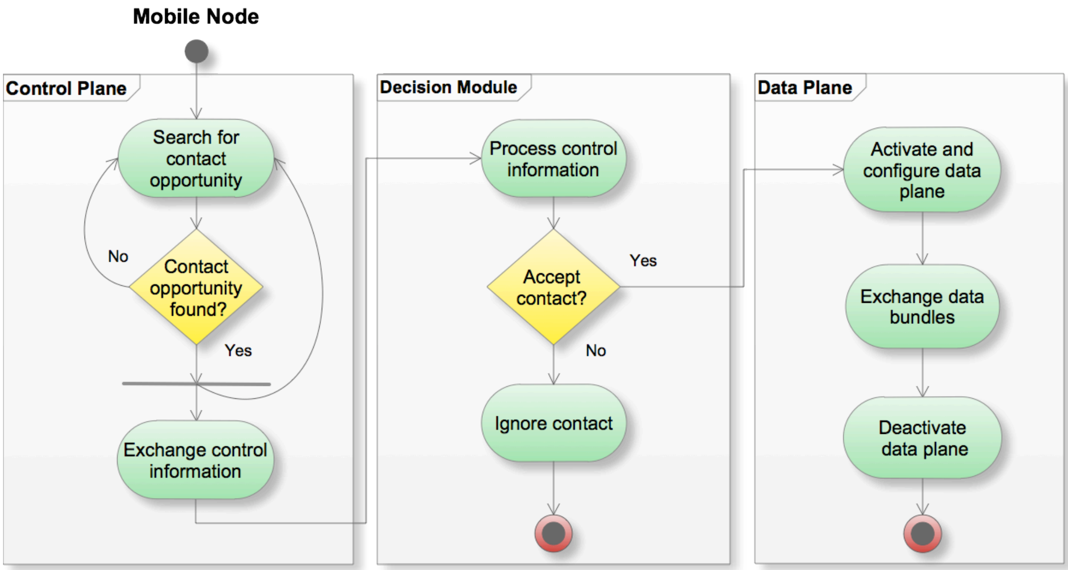

Figure 15. Activity diagram of a mobile node, describing the control and data plane interaction, coordinated by the decision module.

The VDTN testbed used for the network performance evaluations and analysis was created for a laboratory environment. It consists in computers/laptops, portable digital assistants (PDAs), and robots. Desktop and laptop computers are used to emulate the terminal nodes and the relay nodes. Mobile nodes are emulated through LEGO MINDSTORMS NXT robots [60]. A mobile node is shown in Figure 16. As may be seen, each robot carries a PDA having data-networking and storage capabilities. All network nodes support bluetooth [61] and IEEE 802.11b/g [48] technologies. These technologies are used to support the VDTN out-of-band signaling with the separation between control and data planes. While the bluetooth connection is used to exchange signaling information, the 802.11 is used to the transmission of data bundles.

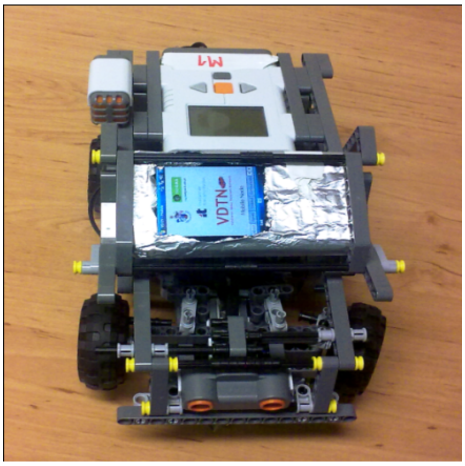

Figure 16. LEGO MINDSTORMS NXT robot with a PDA device coupled.

Figure 17, presents an interaction between two mobile nodes *(a)* and a mobile node with a relay node *(b)*. It is important to notice that LEGO NXT robots, which represent the mobile nodes (e.g. vehicles), are programmed with several mobility models (e.g. random movement across roads, bus movement), introducing different movement patterns.

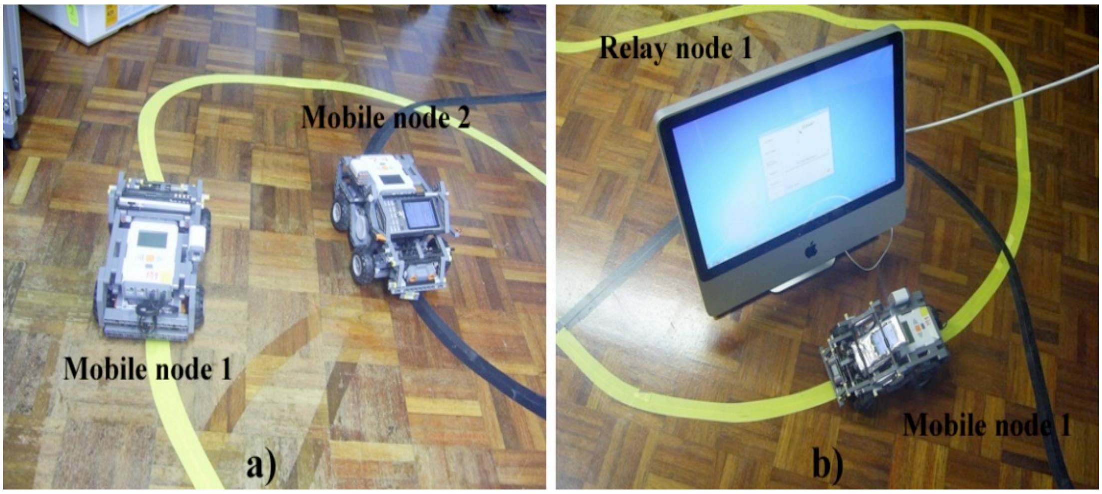

Figure 17. Mobile node 1 interacting with mobile node 2 (a); Mobile node 1 interacting with relay node 1 (b).

This VDTN testbed includes three terminal nodes, four mobile nodes and two relay nodes. Terminal nodes are placed at different points (edges) of the laboratory. Terminal nodes acts as network access points, and it is assumed that one of them is connected to the Internet. Mobile nodes follow different pre-defined paths in order to emulate bus routes. *Mobile node 1* follows a yellow path, *mobile node 2* and *4* follow a black path and *mobile node 3* a white path. Mobile nodes move around the network following their paths and collecting traffic messages with the purpose of replicate these messages through the network. Relay nodes are placed on the intersections of mobile nodes paths in order to increase contact opportunities, collecting and transfer bundles from communications with mobile nodes.

Figure 18, illustrates the class diagram for the VDTN architecture development. The highlighted classes represent the classes used in the implementation of the CSR. This diagram represents the structure and relations between all classes and serves as model for the objects. In the diagram shown below, attributes and methods where omitted to improve its readability.

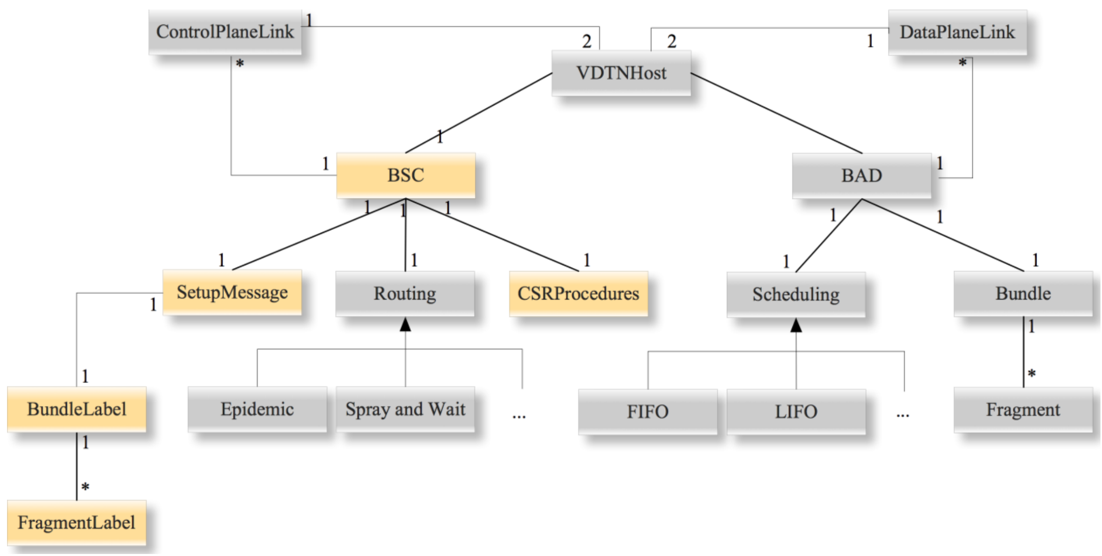

Figure 18. Class diagram of VDTN architecture with CSR mechanisms implemented.

4. Performance Evaluation and Validation

This chapter focuses on the performance evaluation and validation of the CSR mechanisms. Using a described network scenario, firstly it presents the validation and feasibility of the proposal and the performance assessments for Epidemic and Spray and Wait (normal and binary variant) protocols. These evaluations include the comparison between protocols and their impact on the performance of a VDTN network, with and without CSR mechanisms. This study was conducted through the above-presented VDTN testbed.

4.1 Network Scenario

The network scenario includes three terminal nodes, four mobile nodes and two relay nodes. Each mobile node follows its own route as illustrated in Figure 19. Relay nodes are placed in the paths intersection helping the mobile nodes to bypass data along the network. In the presented application scenario, *terminal node 1* furnishes several services to the network, answering the requests issued by *terminal nodes 2 and 3*. These services include text, image and video files. *Terminal node 1* also generates data bundles with temperature of its computer CPU to be delivered and stored on terminal nodes 2 and 3. It is assumed that during data transfer between nodes, there is no bundle fragmentation. Incomplete or loss transfers are considered as a failed data exchange and re-transmitted in a next contact opportunity.

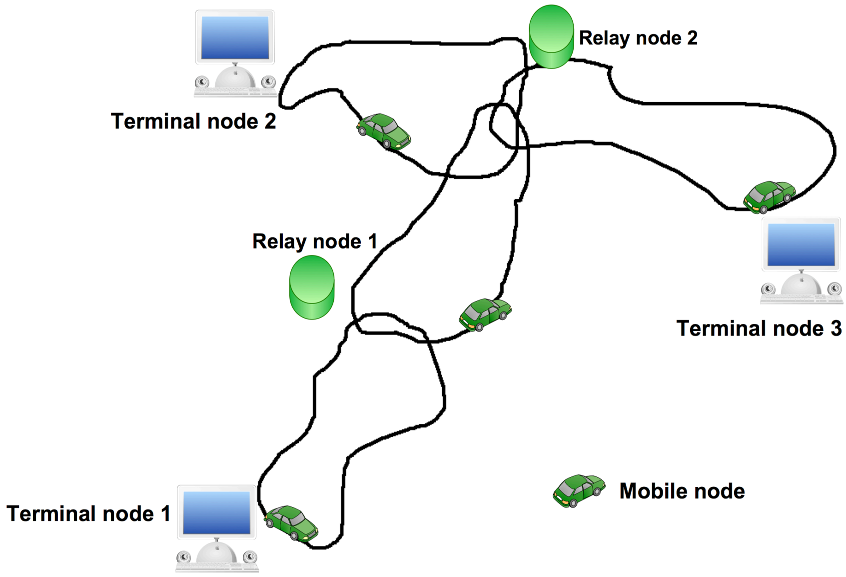

Figure 19. Illustration of the VDTN testbed with three terminal nodes, two relay nodes, and four mobile nodes.

Figures 20,21 and 22 present the VDTN testbed laboratory scenario and all the above-mentioned interactions and behaviors.

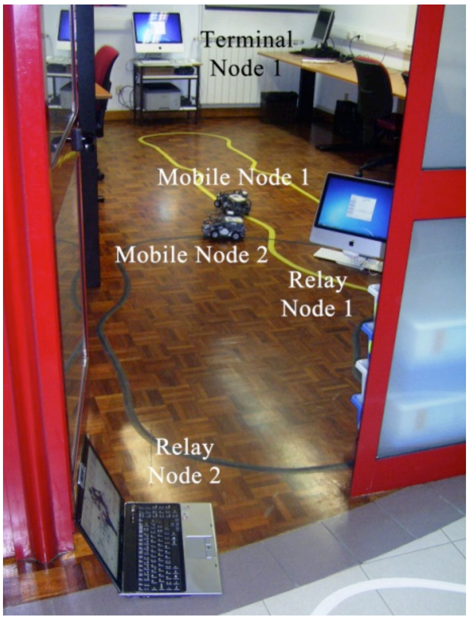

Figure 20. VDTN testbed scenario and the respective interactions between terminal node 1, mobile nodes 1 and 2, and relay nodes 1 and 2.

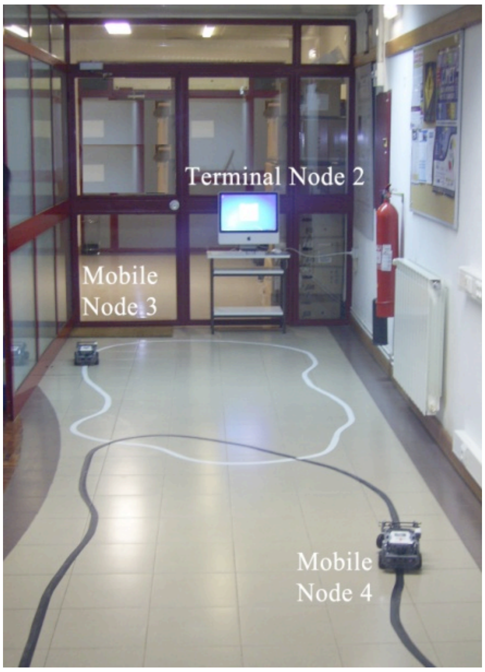

Figure 21. VDTN testbed scenario and the respective interactions between terminal node 2, mobile nodes 3 and mobile node 4.

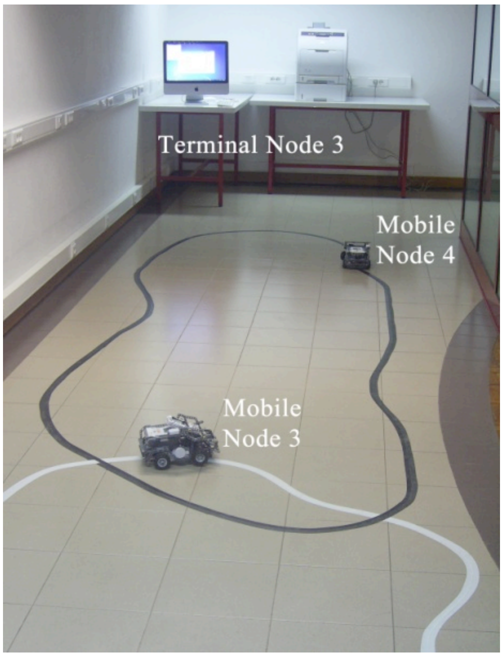

Figure 22. VDTN testbed scenario and the respective interactions between terminal node 3, mobile nodes 3 and mobile node 4.

The performance of the VDTN testbed was evaluated with a bundle generation at every 15, 30 and 60 seconds. Dropping policies assume a first-in, first-out (FIFO) queue where bundles are deleted when their buffer reaches maximum capacity. For dropping purposes, a time-to-live (TTL) field was also included, and bundles are deleted when it expires. TTL changes between 5 and 15 minutes, along tests. Performance metrics considered in this study are the bundle delivery probability, and the bundle average delay (in seconds) that consists in the time between bundle creation and delivery and the average contact duration (seconds) between nodes. In the context of this work, the VDTN testbed has a limited and challenged scenario with low number of network nodes and messages directly. These aspects may affect the network behavior and its performance.

The following two well-known DTN routing protocols are considered: Epidemic [62] and Spray and Wait (including its binary variant) [63]. Epidemic is a flooding based routing protocol where network nodes exchange all the data that they do not have. In an infinite buffer space environment, this protocol performs better than all the other ones. It has the best delivery and latency ratios providing an optimal routing solution. However, in an environment with limited resources, the overall performance severely decreases, due to the flooding that wastes unmeasured network resources. Spray and Wait protocol creates a number of copies *N* to be exchanged ("sprayed") per message. A network node forwards de *N* copies to the first different nodes encountered. Then, it "waits" until one of the network nodes reaches the destination. In a large range of scenarios, Spray and Wait presents better overall network performance than epidemic routing, especially in average message delay and number of transmissions per message delivered [63]. In the binary variant, a node *A* that has more than *1* message copies encounters a node *B* that does not have any copy. Then node *A* forwards to node *B* the number of *N/2* message copies and keeps the rest of the messages. A node with *1* copy left, only forwards it to its final destination.

For assessments and validation purposes of the CSR proposal the following worst-case scenario was assumed: A contact between a relay node with 2 GBytes of stored data and a mobile node. It is assumed that a source node generates one fragment with 2028 KBytes every 7 seconds, and an average number of 4 fragments per bundle. A new VDTN data bundle is generated every 30 seconds (following time based assembly algorithm)[]. The size of a CSR bundle fragment label is 192 bits (equivalent to 0,096 KBytes per bundle fragment). Therefore, it is estimated that 2 GBytes of data in the relay node stores an average of 256 bundles with 8192,12 KBytes of size (data and labels) each one. Then for each 2GBytes of data, 24,576 Kbytes are CSR labels. Through the VDTN testbed it was measured that 1 KB takes approximately 0,11 seconds to be transferred. Therefore, CSR labels of 2 Gbytes of stored data, take approximately 2,64 seconds to be transferred, resulting in very low overhead. This overhead time is perfectly adequated to the speed of any moving vehicle in the VDTN network and, therefore, proves the feasibility of the proposed CSR mechanisms.

4.2 Performance Analysis

This section presents the VDTN network behavior and performance with and without CSR mechanisms. It presents a performance evaluation on both above-mentioned routing protocols, Epidemic and Spray and Wait (including its binary variant).

4.2.1 Performance Evaluation of Epidemic Routing Protocol

Figure 23 shows the bundle delivery probability as function of number of data bundles generated per minute. As may be seen, CSR mechanisms clearly improve the network performance presenting better results on the three different input traffic loads. With the increase of generated bundles per minute, the bundle delivery probability decreases, as expected. However, with CSR mechanisms, Epidemic protocol increases the bundle delivery probably in 17%, 20% and 22%, respectively, improving the delivery probability on all traffic loads. This network behavior results from the better performance of relay nodes and also from the decreasing number of exchanged bundles per contact, by eliminating redundant data.

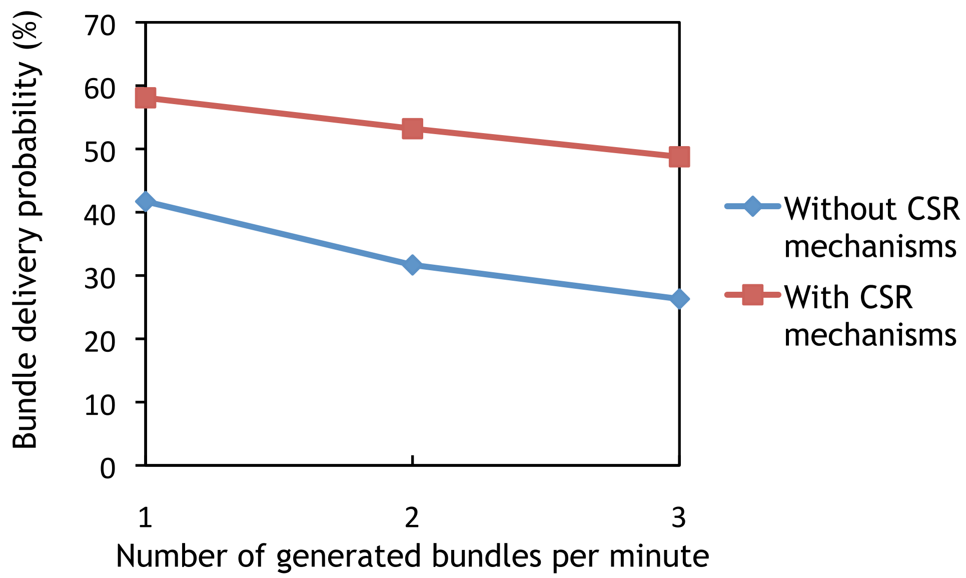

Figure 23. Bundle delivery probability as function of number of generated bundles per minute for Epidemic protocol with and without content storage and retrieval mechanisms.

Figure 24 depicts a data bundle average delay (in seconds) from terminal 1 to terminals 2 and 3. Without CSR mechanisms, bundles that where previously requested or responded are not considered cacheable in any relay node and then, they have to be carried again all around the network until the destination is reached. With CSR mechanisms, these requests and replies are stored in relay nodes, which eventually are closer to destination nodes (terminal nodes), decreasing the average delay. An example is a bundle stored in *relay node 2* containing a content previously requested by *terminal node 3*. If *terminal node 2* issues a request for the same content, then *mobile node 3* encounters it in *relay node 2* instead of sending a request all around the network that "hopefully" reaches *terminal node 1*. With CSR mechanisms on relay node 1, it decreases the data bundle average delay approximately in 123 seconds to terminal node 2 and 120 seconds to terminal node 3. On relay node 2, which is the closest one to the terminal nodes, CSR mechanisms decreases the data bundle average delay approximately in 187 seconds to terminal node 2 and 195 seconds to terminal node 3.

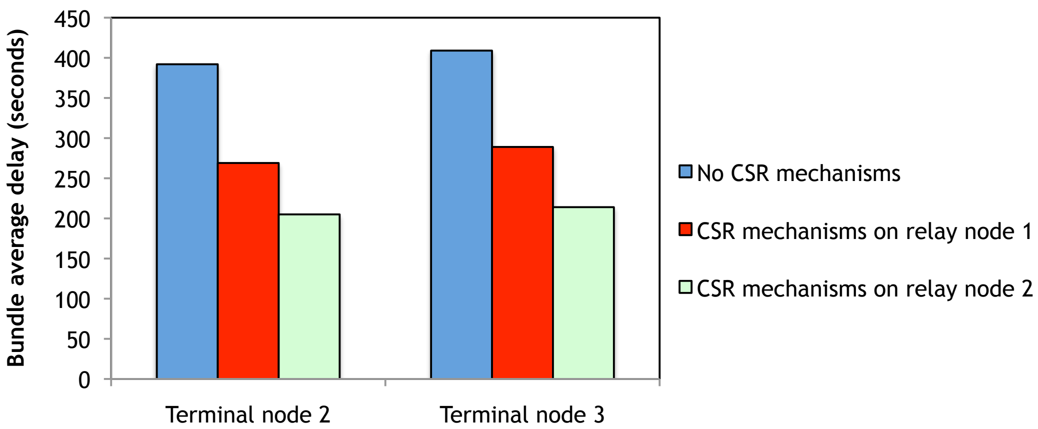

Figure 24. A bundle average delay from terminal 1 to terminals 2 and 3 with and without content storage and retrieval mechanisms for Epidemic protocol.

The bundle average delay improves the network performance significantly with CSR mechanisms. Figure 25, presents the bundle average delay (in seconds) as function of number of generated bundles per minute with and without content storage and retrieval mechanisms. The bundle average delay benefits from CSR mechanisms, decreasing the delay in 80, 106 and 116 seconds, respectively. These performance improvements also benefits from the relay nodes storage and forwarding procedures presented on Figure 24. Relay nodes store previous requested bundles that are forwarded every time that the same requests are issued from terminal nodes, that eventually are closer to the relay node than the source node.

Figure 26 presents the average contact duration between nodes as function of number of generated bundles per minute. Due to vehicles mobility, contact between nodes must be brief and fast to guarantee that if not all, the biggest amount of data is exchange. CSR mechanisms clearly improve this performance metric through the selection of specific cacheable and forwarding contents eliminating redundant data. These procedures eliminate all the exchanged traffic waste, saving contention resources (e.g. buffer space) that also directly affect the bundle delivery probability.

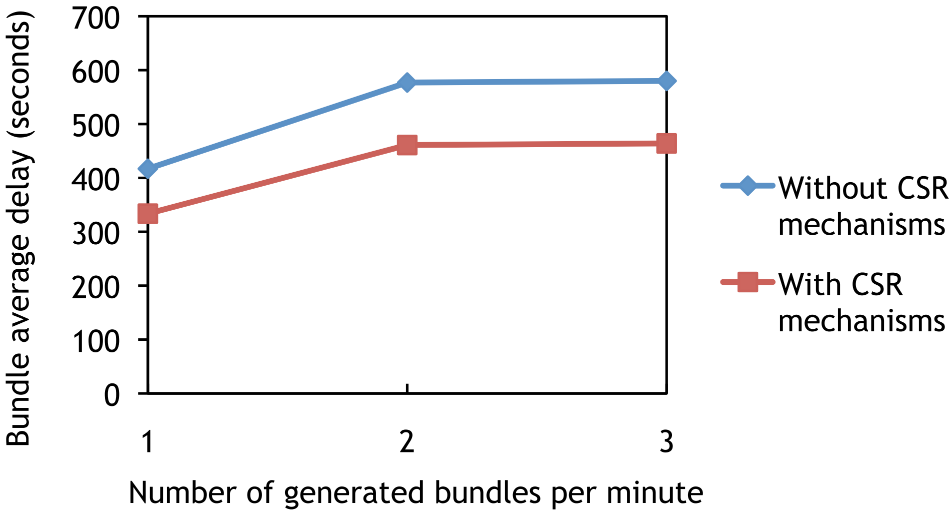

Figure 25. Bundle average delay (seconds) as function of number of generated bundles per minute for Epidemic protocol with and without content storage and retrieval mechanisms.

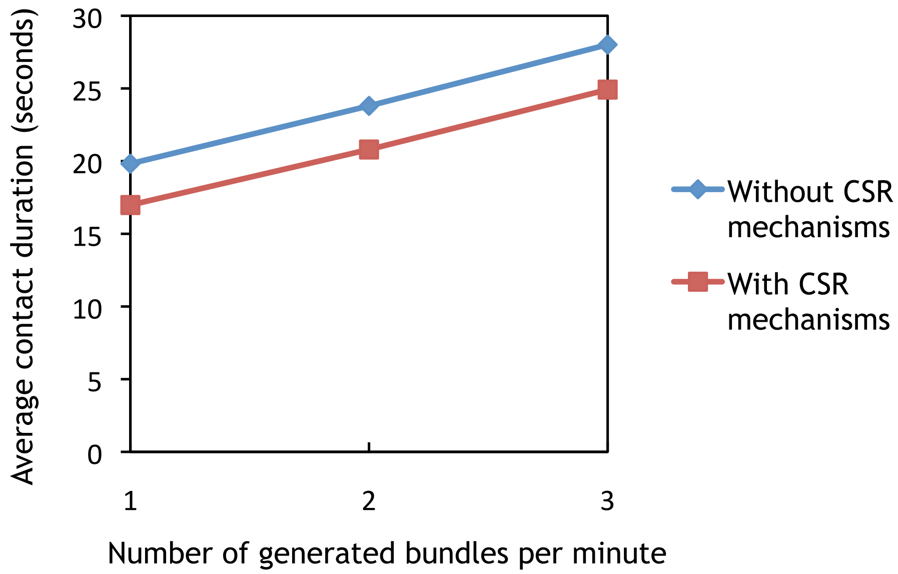

Figure 26. Average contact duration (seconds) as function of number of generated bundles per minute for Epidemic protocol with and without content storage and retrieval mechanisms.

4.2.2 Performance Assessment of Spray and Wait Routing Protocol

Figure 27, shows the bundle delivery probability as function of number of generated bundles per minute for Spray and Wait protocol. It is expected that network performance with Spray and Wait routing protocol may be better than Epidemic in all performance metrics. However, as may be seen Epidemic presents a better performance. Nevertheless, CSR mechanisms increase significantly the bundle delivery probability for all input loads. Figure 28, presents the bundle average delay as function of number of generated bundles per minute for Spray and Wait protocol. It is shown that Spray and Wait performs better than Epidemic. With CRS mechanisms, the average delay decreases 124, 163 and 180 seconds for each class of input traffic load, respectively. In the Spray and Wait protocol this metric reduces even more the time that a bundle is carried through the network, reducing also the contention and waste of resources (buffer space).

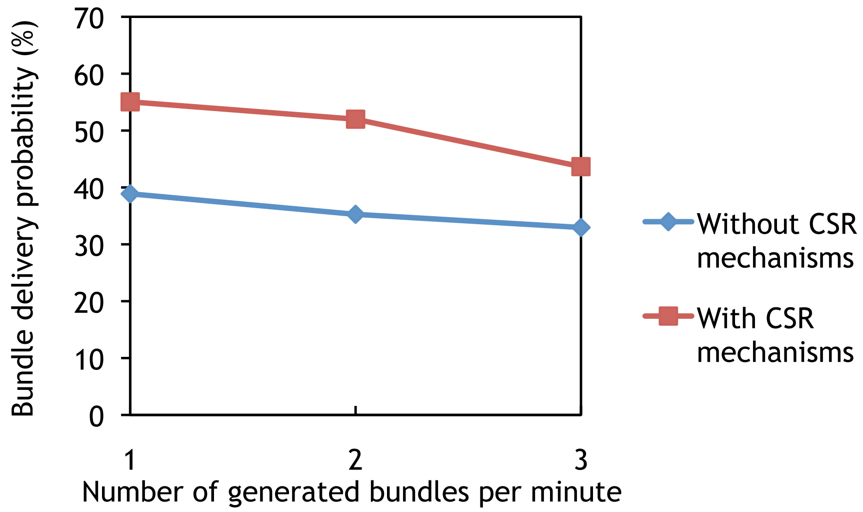

Figure 27. Bundle delivery probability as function of number of generated bundles per minute for Spray and Wait protocol with and without content storage and retrieval mechanisms.

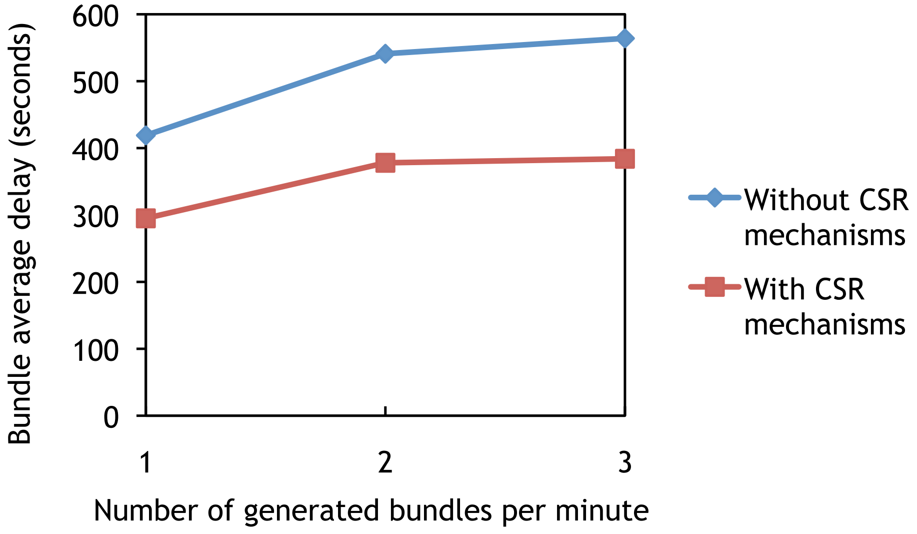

Figure 28. Bundle average delay (seconds) as function of number of generated bundles per minute for Spray and Wait protocol with and without content storage and retrieval mechanisms.

The average contact duration as function of number of generated bundles per minute for Spray and Wait protocol is shown in Figure 29. Spray and Wait also presents better results than Epidemic. As may be observed, for Spray and Wait, CSR mechanisms decreases even more the contact duration between nodes and the average contact duration decreases about 9, 12 and 10 seconds, respectively, for the three input loads.

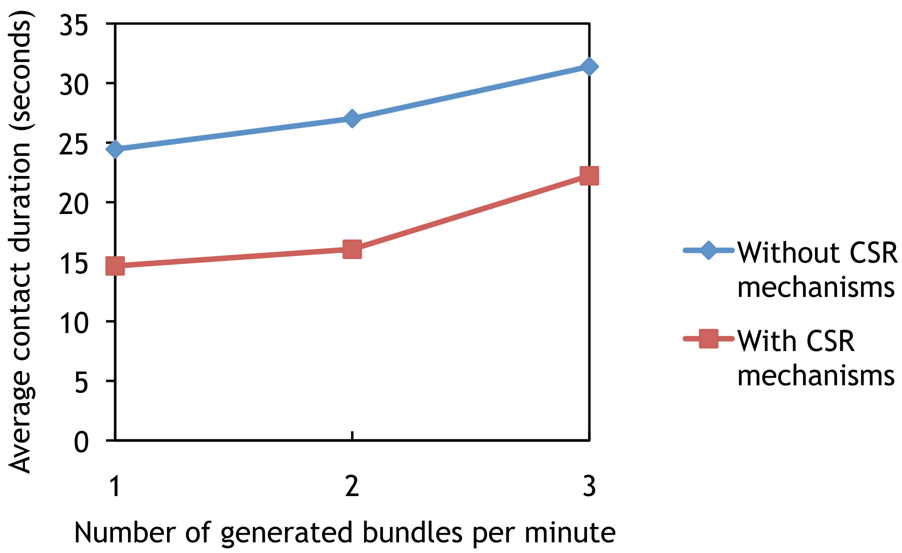

Figure 29. Average contact duration (seconds) as function of number of generated bundles per minute for Spray and Wait protocol with and without content storage and retrieval mechanisms.

4.2.3 Performance Assessment of Spray and Wait Routing Protocol in Binary Variant

Figure 30, shows the bundle delivery probability as function of number of generated bundles per minute for Spray and Wait protocol in its binary variant. It is expected that network performance with Spray and Wait binary mode may be better than Epidemic and Spray and Wait, in all performance metrics. However, Epidemic protocol still presents a better performance on bundle delivery probability than both variants of Spray and Wait. Nevertheless, CSR mechanisms increase significantly the bundle delivery probability for all input loads. The delivery probability with CSR mechanisms increases 4,56%, 0,45% and 3,33% respectively. The performance improvements from the binary to the normal variant of Spray and Wait protocol are very small and with the increase of traffic load it tends to equalize on both protocol variants.

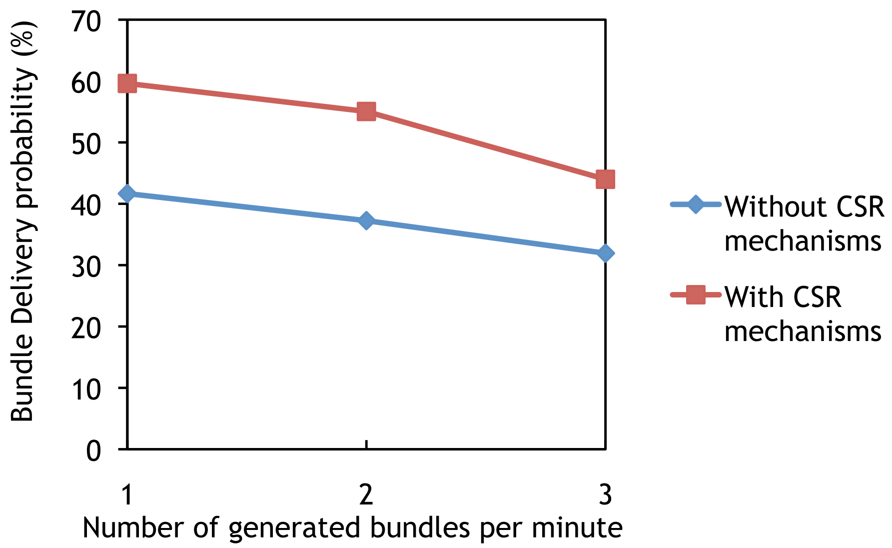

Figure 30.Bundle delivery probability as function of number of generated bundles per minute for spray and wait protocol (binary variant) with and without content storage and retrieval mechanisms.

Figure 31 and 32, presents the bundle average delay and the average contact duration as function of number of generated bundles per minute for Spray and Wait protocol in its binary variant. It is shown that in this binary variant, the Spray and Wait protocol achieves better performs than the normal variant and Epidemic protocol. In both metrics, by increasing the number of generated bundles per minute the Spray and Wait binary mode values are very similar to the obtained values in the normal variant. With CRS mechanisms, the average delay decreases 125,2, 160,3 and 182,7 seconds for each class of input traffic load, respectively.

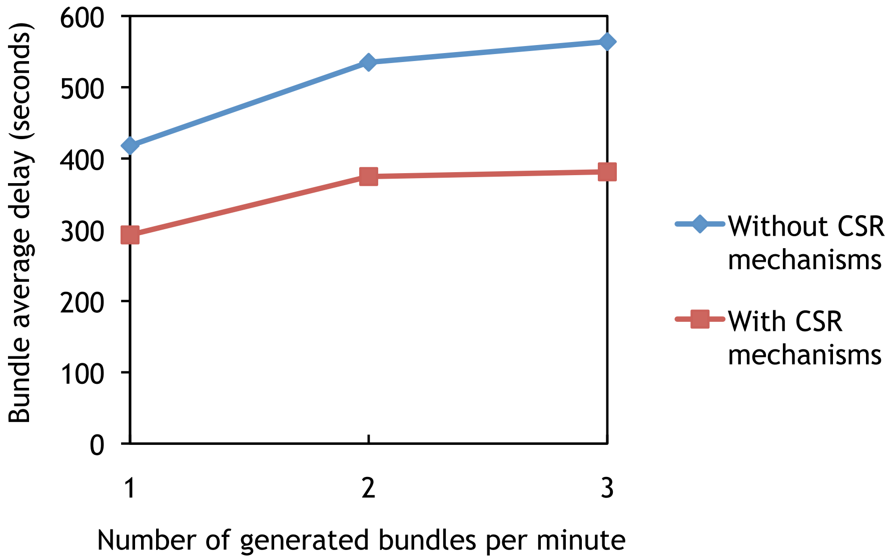

Figure 31.Bundle average delay (seconds) as function of number of generated bundles per minute for spray and wait protocol (binary variant) with and without content storage and retrieval mechanisms.

As may be observed in Figure 32, for the average contact duration (seconds) between nodes decreases about 6, 11,7 and 10,14 seconds, respectively, for each class of input traffic load, respectively.

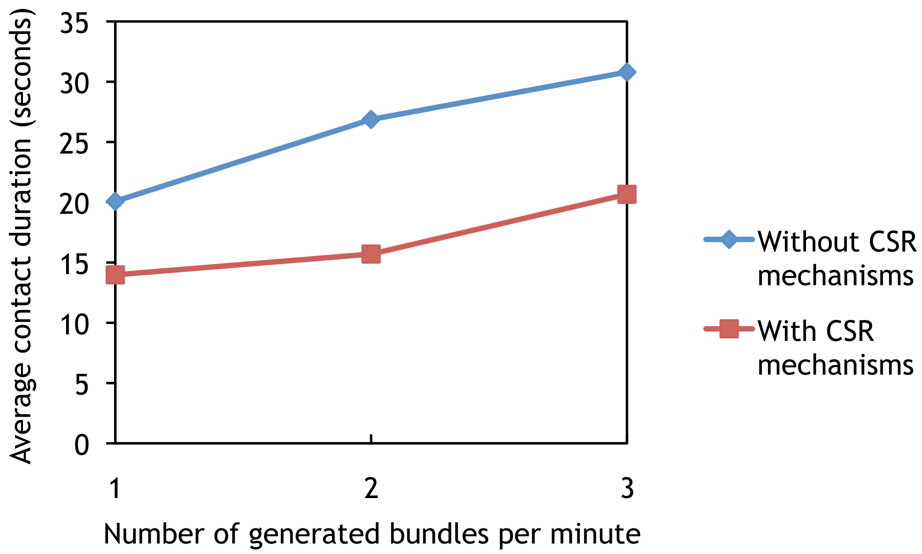

Figure 32. Average contact duration (seconds) as function of number of generated bundles per minute for spray and wait protocol (binary variant) with and without content storage and retrieval mechanisms.

5. Conclusions and Future Work

This Chapter presents a synthesis of the main achievements and points to several directions for future work. After introducing and presenting the topic of this dissertation and define its objectives, Chapter 2 presented the revision on VDTN literature and review of the state of the art on caching and forward related projects. This review focuses on Web-caching schemes, ad-hoc networks caching mechanisms, and mainly on DTNs storage and forward solutions. Afterwards, the CSR proposal and all its key features and characteristics are described.

Chapter 3 presented the design and application of the CSR mechanisms on VDTNs. The conceptual design for the proposal was described. Furthermore, this Chapter presents the construction of data control algorithms responsible for caching and forward operations, resulting in the intelligent cashing solution needed to improve the performance of our VDTN network. Finally, the VDTN testbed used for network performances and evaluations is exposed introducing its main features, equipment and used technologies.

Chapter 4 presented the performance assessments of the CSR mechanisms, using the VDTN testbed. For these performance evaluations, different network setups were used. Epidemic and Spray and Wait (source and binary mode) routing protocols, and different input traffic loads were applied and performed on the testbed, comparing and analyzing the network performance with and without CSR mechanisms. The performance metrics considered were the bundle delivery probability, the bundle average delay (in seconds) and the average contact duration (seconds) between nodes. This metrics were measured and compared as function of number of generated bundles, which increases from 1 to 3 bundles per minute.

Through the network performance evaluation and its metrics comparisons, with and without CSR mechanisms on both routing protocols, it was possible to conclude that the CSR solution have improved significantly the overall performance of VDTN networks. Despite of some results variations due to the testbed challenged and limited laboratory environment, CSR mechanisms have shown significantly improvements on the bundle delivery probability, the average bundle delay, and on the average contact duration between nodes for both implemented protocols.

The main objective of this thesis was to present a content storage and retrieval solution for VDTNs. To accomplish this main objective, the related work on VDTNs and caching and forward related approaches has been studied and presented. The data control algorithms and intelligent cashing proposal where design and constructed, resulting in the CSR mechanisms for VDTN. The feasibility and performance evaluation was successfully measured through the application of CSR mechanisms on the VDTN testbed.

All the proposed objectives where successfully accomplished.

5.1 Future Works

To conclude this work, it just remains to suggest future research directions based on current work:

- More application scenarios (e.g., file transfer protocol or World Wide Web applications for VDTN).

- Performance evaluation and analysis of other routing protocols using CSR mechanisms.
- Performance evaluation of CSR mechanisms through simulation.
- Evaluation in a larger scale with more testbed nodes, and a demonstration with real vehicles are also part of our future research interests.

References

[1] V. N. G. J. Soares, F. Farahmand, and J. J. P. C. Rodrigues, "A Layered Architecture for Vehicular Delay-Tolerant Networks," in *IEEE Symposium on Computers and Communications* (ISCC 2009), Sousse, Tunisia, 2009, pp. 122-127.

[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," in *Applications, Technologies, Architectures, and Protocols for Computer Communication (SIGCOMM)* Karlsruhe, Germany, August, 2003, pp. 27-34.

[3] S. Corson and J. Macker, "Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations," in *RFC: 2501*, 1999.

[4] L. Buttyán and J.-P. Hubaux, "Stimulating cooperation in self-organizing mobile ad hoc networks," *Mobile Networks and Applications,* vol. 8, pp. 579-592, 2003.

[5] A. Vasiliou and A. A. Economides, "Rescue operations using coordinated multicast MANET," in *Fifth International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP'06),* Patras, Greece, July 2006, pp. 1-6.

[6] X. Lu, Y.-C. Chen, P. Liò, and D. Towsley, "A Novel Mobility Model from a Heterogeneous Military MANET Trace," *Computer Cience*, vol. 5198, pp. 463-474, 2008.

[7] M. A. Yousuf, N. Begum, B. Sultana, and N. Amin, "Concept of Mobility Based Video-Telephony in Mobile Ad Hoc Network (Manet)," *Asian Journal of Informational Technology* vol. 5, pp. 829-833, 2006.

[8] S. Yousefi, M. S. Mousavi, and M. Fathy, "Vehicular Ad Hoc Networks (VANETs): Challenges and Perspectives," in *6th International Conference on ITS Telecommunications Proceedings*, Chengdu, China, June 2006, pp. 761-766.

[9] Research and Innovative Technology Administration (RITA), "Intelligent Transportation Systems," [Online]. Available: http://www.its.dot.gov/index.htm. [Accessed: March, 2010].

[10]Y. Lin, Y. Chen, and S. Lee, "Routing Protocols in Vehicular Ah Hoc Networks: A Survey and Future Perspectives," *Journal of Information Science and Engineering*, vol. 26, No. 3, pp. 913-932, 2010.

[11]I. Leontiadis and C. Mascolo, "GeOpps: Geographical Opportunistic Routing for Vehicular Networks," in *IEEE International Symposium on a World of Wireless, Mobile and Multimedia Networks, 2007 (WoWMoM 2007)* Espoo, Finland, June 2007, pp. 1-6.

[12]O. Brickley, C. Shen, M. Klepal, M. Tabatabaei, and M. Pesch, "A Data Dissemination Strategy for Cooperative Vehicular Systems," in *Vehicular Technology Conference 2007 (VTC 2007)*, Dublin, Ireland, April 2007, pp. 2501-2505.

[13]J. Ott and D. Kutscher, "Drive-thru Internet: IEEE 802.11b for "automobile"," in *INFOCOM 2004*, vol. 1, pp. 362-373, 2004.

[14]K. Lee, S. Lee, R. Cheung, and U. Lee, "First experience with cartorrent in a real vehicular ad hoc network testbed," in *Mobile Networking for Vehicular Environments*, Achorage, Alaska, USA, May 2007, pp. 109-114.

[15]R. Tatchikou, S. Biswas, and F. Dion, "Cooperative vehicle collision avoidance using inter-vehicle packet forwarding," in *Global Telecommunications Conference, 2005. GLOBECOM '05*. vol. 5, St. Louis, Missouri, USA, December 2005, pp. 2766.

[16]L. Franck and F. Gil-Castineira, "Using Delay Tolerant Networks for Car2Car communications," in *IEEE International Symposium on Industrial Electronics, 2007(ISIE 2007)*, Vigo, Spain, June 2007, pp. 2573-2578.

[17] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, "Delay-Tolerant Networking Architecture," *RFC 4838*, April 2007.

[18]K. Fall, S. Farrell, "DTN: an architectural retrospective", IEEE Journal on Selected Areas in Communications, vol.26, No.5, pp. 828-836, June 2008.

[19]K. Scott, S. Burleigh, "Bundle Protocol Specification", IETF RFC 5050, November 2007.

[20]S. Burleigh, A. Hooke, L. Torgerson, K. Fall, V. Cerf, B. Durst, K. Scott, and H. Weiss, "Delay-Tolerant Networking: An Approach to Interplanetary Internet," *IEEE Communications Magazine*, vol. 41, pp. 128-136, June 2003.

[21]J. Partan, J. Kurose, and B. N. Levine, "A Survey of Practical Issues in Underwater Networks," in *1st ACM International Workshop on Underwater Networks, in conjunction with ACM MobiCom 2006* Los Angeles, CA, USA, September 2006, pp. 17-24.

[22]P. Juang, H. Oki, Y. Wang, M. Martonosi, L. S. Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," *ACM SIGOPS Operating Systems Review,* vol. 36, December 2002.

[23]A. Pentland, R. Fletcher, and A. Hasson, "DakNet: rethinking connectivity in developing nations," *Computer,* vol. 37, pp. 78-83, January 2004.

[24]J. J. P. C. Rodrigues, V. N. G. J. Soares, and F. Farahmand, "Stationary Relay Nodes Deployment on Vehicular Opportunistic Networks," in *Mobile Opportunistic Networks: Architectures, Protocols and Applications*, M. K. Denko, Ed. USA: Auerbach Publications, CRC Press, 2010.

[25]Joel J. P. C. Rodrigues, Vasco N. G. J. Soares, and Farid Farahmand, "Stationary Relay Nodes Deployment on Vehicular Opportunistic Networks", in *Mobile Opportunistic Networks*: *Architectures, Protocols and Applications*, Mieso K. Denko (Ed.), Auerbach Publications, CRC Press, USA, ISBN: 978-1420088120, May 2010

[26]J. A. Dias, J. N. Isento, B. M. Silva, V. N. G. J. Soares, P. Salvador, A. Nogueira, and J. J. P. C. Rodrigues, "Creation of a Vehicular Delay-Tolerant Network Prototype," in *Engineering 09: Innovation and Development*, Covilhã, Portugal, November 2009.

[27]V. N. G. J. Soares, J. J. P. C. Rodrigues, F. Farahmand, and M. Denko, "Exploiting Node Localization for Performance Improvement of Vehicular Delay-Tolerant Networks," in *2010 IEEE International Conference on Communications (IEEE ICC 2010) - General Symposium on Selected Areas in Communications (ICC'10 SAS)*, Cape Town, South Africa, May 2010.

[28]N. Banerjee, M. D. Corner, and B. N. Levine, "An Energy-Efficient Architecture for DTN Throwboxes," in *26th IEEE International*

Conference on Computer Communications (INFOCOM 2007), Anchorage, Alaska, USA, May 2007, pp. 776-784.

[29]J. J. P. C. Rodrigues, *Optical Burst Switching Networks Architectures and Protocols*, Fundação Nova Europa (Ed.), University of Beira Interior, Covilhã, Portugal, 2008.

[30]A. Luotonen and K. Altis, "World Wide Web proxies," in *Computer Networks and ISDN Systems, First International Conference on WWW*, 1994.

[31]Abhijit Gadkari (October 2008), Caching in the Distributed Environment [Online]. Available: http://msdn.microsoft.com/en-us/library/dd129907.aspx.[Accessed: May 2010]

[32]J. Wang, "A survey of web caching schemes for the internet," *ACM Computer Communication,* vol. 29, pp. 36-46, 1999.

[33]V. Issarny and F. Sailhan, "Energy-aware Web Caching for Mobile Terminals," in *Proceedings of the CDCS Workshop on Web Caching Systems*, Vienna, Austria, July 2002, pp. 820-825.

[34]Ircache Project [Online]. Available: http://www.ircache.net. [Accessed : Dezember 2009].

[35] National Laboratory for Applied Network Research [Online]. Available: http://www.nlanr.net.[Accessed: December 2009].

[36]D. Wessels and K. Claffy, "Internet Cache Protocol (ICP)," in *RFC 2186 Version 2* US, 1997.

[37]A. Chankhunthod, P. B. Danzig, C. Neerdaels, M. F. Schwartz, and K. J. Worrel, "A hierarchical Internet object cache," in *USENIX 96* San Diego, CA, USA, 1996.

[38]J. Yang, W. Wang, R. Muntz, and J. Wang, "Access driven Web caching," in *UCLA Technical Report*, 1999.

[39]S. Michel, K. Nguyen, A. Rosenstein, L. Zhang, S. Floyd, and V. Jacobson, "Adaptive Web caching: towards a new caching architecture," *Computer Network and ISDN Systems,* vol. 30, November 1998.

[40]Z. Wang and J. Crowcroft, "Cachemesh: a distributed cache system for world wide web," in *Second International Web Caching Workshop*, 1997.

[41]J. Gwertzman and M. Seltzer, "The case for geographical push-caching," in *Fifth Workshop on Hot Topics in Operating Systems, 1995. (HotOS-V 1995),* Orcas Island, WA, USA,1995, pp. 51-55.

[42]Bangnan, X., Hischke, S. and Walke, B. "The role of ad hoc networking in future wireless communications," *International Conference on Communication Technology (ICCT 2003)*, Vol. 2, Beijing, China, April 2003, pp. 1353-1358.

[43]L. Yin, J. Zhao, P. Zhang, and G. Cao, "Cooperative Cache Based Data Access in Ad Hoc Networks," *Computer,* vol. 37, pp. 32-39, 2004.

[44]M. Fiore, F. Mininni, C. Casetti, and C. Chiasserini, "To Cache or Not To Cache?," in *IEEE INFOCOM 2009,* Rio de Janeiro, Brazil, April 2009, pp. 235-243.

[45]Y. Du and S. K. S. Gupta, "COOP - A cooperative caching service in MANETs," in *Autonomic and Autonomous Systems and International Conference on Networking and Services - (icas-icns'05),* Papeete, Tahiti, October 2005, pp. 58-58.

[46]F. Sailhan and V. Issarny, "Cooperative caching in ad hoc networks," in *4th International Conference on Mobile Data Management (MDM)* Melborne, Australia, January 2003, pp. 13-28.

[47]J. Ott and D. Kutscher, "The "drive-thru" architecture: WLAN-based Internet access on the road," in *IEEE 59th Vehicular Technology Conference, (VTC-Spring 2004)*, vol. 5, Milan, Italy, May 2004, pp. 2615-2622.

[48]IEEE 802.11 WLAN standard [Online]. Available: http://standards.ieee.org/getieee802.
[Accessed: December, 2009].

[49]J. Border, M. Kojo, J. Griner, and Z. Shelby, "Performance Enhancing Proxies Intended to Mitigate Link-Related Degradations," *RFC 3135,* June 2001.

[50] S. Isaacman and M. M. tonosi, "Potential for Collaborative Caching and Prefetching in Largely-Disconnected Villages," in *Proceedings of the 2008 ACM Workshop on Wireless Networks and Systems for Developing Regions,* San Francisco, CA, USA, September 2008, pp. 23-30, .

[51]J. Ott and M. J. Pitkanen, "DTN-based Content Storage and Retrieval," in *First WoWMoM Workshop on Autonomic and Opportunistic Communications (AOC),* Helsinki, Filand, June 2007, pp. 1-7.

[52]J. Ott and M. J. Pitkanen, "Enabling opportunistic storage for mobile DTNs," *Pervasive and Mobile Computing,* vol. 4, pp. 579-594, 2008.

[53]J. Ott and D. Kutscher, "Bundling the Web: HTTP over DTN," in *Proceedings of Workshop on Networking in Public Transport 2006 (WNEPT 06),* Waterloo, Ontario, Canada, August 2006.

[54]M. J. Pitkanen and J. Ott, "Redundancy and Distributed Caching in Mobile DTNs," in *Mobility In The Evolving Internet Architecture, Proceedings of 2nd ACM/IEEE international workshop on Mobility in the evolving internet architecture,* Kyoto, Japan, August 2007, pp. 1-7.

[55]Object Management Group, Inc., "Unified Modeling Language (UML)," [Online]. Available: http://www.uml.org/ [Accessed: January, 2010].

[56]Microsoft, "Visual C# Developer Center," [Online]. Available: http://msdn.microsoft.com/en-us/vcsharp/default.aspx [Accessed: January, 2010].

[57]Microsoft, "Microsoft .NET Compact Framework," [Online]. Available: http://msdn.microsoft.com/en-us/netframework/aa497273.aspx. [Accessed: January, 2010].

[58]Microsoft, "Windows Mobile," [Online]. Available: http://www.microsoft.com/windowsmobile/en-us/meet/default.mspx. [Accessed: January, 2010].

[59] “Microsoft Visual Studio PRO 2008,” [Online]. Available: http://msdn.microsoft.com/en-us/vstudio.aspx [Accessed: January, 2010].

[60]The LEGO Group, "LEGO Mindstorms NXT," [Online]. Available: http://mindstorms.lego.com/en-us/default.aspx [Accessed: January, 2010].

[61]Bluetooth Special Interest Group [Online]. http://www.bluetooth.com. [Accessed: December, 2009].

[62]A.Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," in *Duke University* Technical Report CS-2000-06, April, 2000.

[63]T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," in *ACM SIGCOMM*, Philadelphia, USA, August 2005, pp. 252-259.